Zoran Hercigonja, mag.edu.inf.

Primjeri pisanih priprema za izvedbu nastavnog sata iz metodike nastave informatike za osnovne i srednje škole

- 1

ISBN 978-953-8200-15-1

## Impressum

Naslov: Primjeri pisanih priprema za izvedbu nastavnog sata iz metodike nastave informatike za osnovne i srednje škole Autor: Zoran Hercigonja, mag.edu.inf. Lektorica: Adela Brozd Nakladnik: Vlastita naklada autora Mjesto i godina izdavanja: Imbriovec Jalžabetski 45c, 2019. godine Vrsta publikacije: Stručna elektronička knjiga Medij: Mrežna publikacija Format: A4 Br. stranica: 256 Područje: Metodika nastave informatike ISBN 978-953-8200-15-1 drugo izdanje

## Predgovor

Knjiga "Primjeri pisanih priprema za izvođenje nastavnog sata iz metodike nastave informatike za osnovne i srednje škole", sastoji se od detaljnih priprema javnih i individualnih predavanja održanih na sljedećim školama: Elektrostrojarska škola Varaždin, Prva gimnazija Varaždin, Druga osnovna škola Varaždin, Šesta osnovna škola Varaždin te Peta gimnazija Zagreb. Pisane pripreme su primjeri najbolje prakse izvođenja nastavnog sata na spomenutim školama. U njima se nalazi tijek nastavnog sata s mikroplanom i makroplanom te svim metodičkim elementima poput socijalnih oblika rada, tehnike poučavanja, vrste nastave, odgojni i obrazovni zadaci, nastavne metode, pomagala i sredstava te detaljna razrada sata po unaprijed zadanoj strukturi nastavnog sata (uvodni dio, glavni dio i završni dio). Pisane pripreme su namijenjene prvenstveno budućim nastavnicima informatike u osnovnim i srednjim školama, ali i svima polaznicima tečajeva pedagoško, psihološko, metodičkog osposobljavanja za budući rad u školama kao i učiteljima primarnog obrazovanja s pojačanim modulom informatike. U pripremama se nalaze sadržaji iz osnovnih i srednjih škola tako da budući nastavnici mogu napraviti paralelnu usporedbu između pripreme i strukture nastavnog sata u osnovnoj i srednjoj školi te razliku u vođenju sata na tehničkim školama i školama gimnazijskog programa. Također u ovoj knjizi se nalazi i detaljna priprema po kojoj je vođen nastavni sat na Državnom stručnom ispitu na Petoj gimnaziji Zagreb. Sve pripreme su prošle analizu i recenzije metodičara i mentora sunastavnika. Također sve pripreme su primijenjene tijekom izvođenja nastavnog sata i zaživjele su u praksi.

## Sadržaj:

1. Javna predavanja-srednje škole (5)

1.1.Primjer pisane pripreme 1-javno predavanje (6)

1.2. Primjer pisane pripreme 2-javno predavanje (19)

1.3. Primjer pisane pripreme 3-javno predavanje (32)

1.4. Primjer pisane pripreme 4-javno predavanje (48)

1.5.Primjer pisane pripreme 5-javno predavanje (73)

2. Individualna predavanja-srednje škole (90)

2.1.Primjer pisane pripreme 1-individualno predavanje (91)

2.2.Primjer pisane pripreme 2-individualno predavanje (102)

3. Javna predavanja-osnovne škole (118)

3.1.Primjer pisane pripreme 1-javno predavanje (119)
3.2.Primjer pisane pripreme 2-javno predavanje (137)
3.3.Primjer pisane pripreme 3-javno predavanje (146)
3.4.Primjer pisane pripreme 4-javno predavanje (167)

4. Individualna predavanja-osnovne škole (182)

- 4.1.Primjer pisane pripreme 1- individualno predavanje(183)4.2.Primjer pisane pripreme 2-individualno predavanje(200)
- 5. Primjer pisane pripreme zaizvedbu nastavnog sata na stručnom ispitu(216)
  5.1. Primjer obrasca pisane priprave za izvedbu nastavnog sata na stručnom ispitu(217)

5.2.Pisana priprava za izvedbu nastavnog sata na stručnom ispitu(225)

## 1. Javna predavanja-srednje škole

Pisane pripreme za izvođenje nastavnog sata započinju javnim predavanjima u srednjim školama. Pripreme su pisane za dvije srednje škole: Elektrostrojarska škola Varaždin i Prva gimnazija Varaždin. Sadržaj priprema obuhvaća različite obrazovne programe koji se razlikuju između škola (između gimnazijskog i tehničkog programa). Promišljanje nastavnog sata, izrada okvirnog scenarija, zavisi o obrazovnim ishodima svakog nastavnog sata i obrazovnog programa. Jako je dobar primjer tehnička škola (Elektrostrojarska škola) koja ima više obrazovnih tehničkih programa, gdje se uči pojedino područje informatike kao što je multimedija i dizajn odnosno web programiranje ili samo programiranje.

Treba naglasiti da izvođenje nastavnog sata kao i pisanje pripreme treba prilagoditi kako području informatike tako i obrazovnim ishodima određenog tehničkog usmjerenja. U pripremama za izvođenje nastavnog sata, nalaze se i konkretni sadržaji primijenjeni tijekom izvođenja nastavnog sata kao i reference na sadržaje prezentacija i drugih dokumenata korištenih tijekom izvođenja nastavnog sata koji se zbog svoje opsežnosti ne nalaze unnutar ove knjige. Radi zaštite osobnih podataka, imena mentora-nastavnika i metodičara su maknuti iz priprema. Važno je znati da se ovdje nalaze primjeri dobre prakse. Pisana priprema je individualno osmišljavanje nastavnog sata tako da ne treba čuditi što poneka priprema nema mikroplan ili neki drugi element poput navedenih nastavnih sredstava ili pomagala. U narednim stranicama, prikazane su pripreme u cjelosti i originalu kao na dan održavanja nastavnog sata. Javnom predavanju su prisustvovali metodičar, nastavnik-mentor te studenti koji su zajedno nakon izvođenja nastavnog sata analizirali nastavni sat i pisanu pripremu.

## **1.1.** Primjer pisane pripreme **1**. – javno predavanje

Javno predavanje je održano na Elektrostrojarskoj školi Varaždin u 2 razredu usmjerernja multimedijalni tehničar. Nastavni sat se izvodio u radu na aplikaciji Adobe Illustrator na temu Vektorske grafike.

#### Početak pripreme:

Škola: Elektrostrojarska škola, Varaždin

Razred: 2. razred

Usmjerenje: Multimedijalni tehnčar

Nastavna jedinica: Rad s kistovima

Nastavna cjelina: Vektorska grafika (Adobe Illustrator)

**Tip sata:** obrada novog gradiva

Blok sat: Da

#### Cilj nastavnog sata:

Upoznati učenike s mogućnostima i funkcionalnostima kistova u aplikaciji Adobe Illustrator

#### Materijalni zadaci:

- Usporediti pojedine mogućnosti i funkcionalnosti aplikacije Adobe Illustrator
- Razlikovati mogućnosti i funkcionalnosti aplikacije Adobe Illustrator
- Razlikovati četiri tipa kistova: Umjetnički (Art), kaligrafski (Calligraphic), kistovi uzoraka (Pattern), čekinjasti (Bristle).
- Izraditi crtež pomoću PaintBrush tools.
- Prilagoditi postavke kistova.
- Napraviti nove individualne kistove pomoću uzoraka.
- Napraviti crtež s Blob Brush tool.

#### Funkcionalne zadaci:

- Razvijati dobru orijentaciju unutar okruženja Adobe Illustratora
- Osposobljavati učenike za lakši rad s naredbama i funkcijama Adobe Illustratora
- Poticati kreativan pristup rješavanju konkretnog zadatka u aplikaciji Adobe Illustrator
- Poticati interes za samostalno otkrivanje ostalih mogućnosti aplikacije Adobe Illustratora

#### Odgojni zadaci:

- Stvoriti osjećaj za estetski lijepim
- Razvijati smisao za red i organizaciju sadržaja
- Stjecati pozitivan stav prema radu
- Razvijati suradnju i međusobno pomaganje

#### Nastavna pomagala:

- Računalo
- Projektor

#### Nastavne metode:

- Metoda razgovora
- Metoda usmenog izlaganja
- Metoda demonstracije
- Metoda praktičnog rada

#### Oblici rada:

- Frontalni rad
- Samostani rad

#### Literatura:

Adobe Illustrator CS5, (2010.) ClassRoom in a book; The official training workbook from Adobe Systems; Lession 11: Working with brushes, ISBN-10: 032170178X

#### TIJEK IZVOĐENJA NASTAVE

#### Uvod (7 minuta)

N: Dobar dan! Moje je ime X. X. i danas ću biti vaš nastavnik. Molio bih vas da tijekom rada komuniciramo na način da podignete ruke i tek kada budete prozvani postavite svoje pitanje. Prije postavljenog pitanja recite svoje ime radi kvalitetnije komunikacije. Ukoliko će vam se činiti u nekom trenutku da idem prebrzo, prekinite me uz podizanje ruke i ja ću usporiti. Na vrijeme ćete biti obaviješteni kada dođe vrijeme odmora. Na kraju zadnjeg sata ćete ugasiti računala i radno mjesto ostaviti urednim.

Razredom ću poslati dva USB-a s kojih ćete preuzeti na vaša računala datoteku Lession11 sa svim sadržajima. Kad preuzmete sadržaje pošaljite USB drugim učenicima. U sadržajima se nalaze i detaljne upute (Upute za rad u Adobe Illustratoru) koje možete imati otvorene tijekom rada.

#### PONAVLJANJE:

N: Nedavno ste s profesoricom Ljepava radili u Photoshopu, a mi danas radimo u Illustratoru. Kakav zapis koristi Photoshop, a kakav Illustrator?

U: Photoshop koristi rasterski zapis, a Illustrator vektorski zapis

N:Odličan odgovor!

N: Dobro, a što mislite da li je Paint alat za rasterski ili vektorski zapis?

U:Rasterski.

N:Po čemu razlikujemo rasterski zapis od vektorkog zapisa?

U: Na primjer rasterske slike se sastoje od piksela i računalo sprema sliku tako da pamti točan položaj i boju piksela. Vektorski zapis računalo pamti kao matematičke funkcije koje opisuju krivulje.

N: Točan odgovor.

N: U Photoshopu ste radili s kistovima.Čemu uopće kistovi služe?

U: Kistovi služe da bi različitim oblicima mogli stvarati crteže različitih tekstura i oblika.

#### Glavni dio sata: 1. Sat (38 minuta)

U ovom djelu sata, učenicima ću objasniti i demonstrirati vrste kistova te mogućnosti rada s njima vodeći se po koracima koji su navedeni u uputama (Upute za rad u Adobe Illustratoru) i u ponekom trenutku im dati manji zadatak u trajanju od 4-5 minute da na crtežu isprobaju pokazanu i demonstriranu funkcionalnost te na kraju drugog sata dati zadatak za samostalan rad koji će objediniti sve demonstrirane funkcije i mogućnosti rada s kistovima tijekom tog sata.

N: Prvo svi pokrenite aplikaciju Adobe Illustrator.

(Za vrijeme dok oni pokreću aplikaciju, ja učitavam gotovu sliku u aplikaciju Adobe Illustrator kako bi vizualno prezentirao cilj tadašnjeg sata)



N: Cilj današnjih predavanja je upotrebom i kombinacijom kistova te slojeva stvoriti ovakav crtež. U datoteci Lession11 se nalazi dokument L11start\_1.ai. Radi se o nedovršenoj slici koju ćemo zajednički doraditi. Prije početka rada napravite si sigurnosnu kopiju dokumenta L11start\_1.ai radi prevencije i tu istu kopiju otvorite u Adobe Illustratoru. U datoteci Lession 11 imate i rješenje kako treba izgledati crtež.



N: Prvo uključite Layer panel klikom na ikonu 🐳 i pogledajte stanje slojeva. Slojevi su gotovo identični slojevima kao i kod Photoshopa. Možete primijetiti iste opcije (visibility, zaključavanje, otključavanje).

N:Jedna napomena: ako dođe do pogreške u nekom koraku, vratite prethodno stanje sa CTRL+Z i prije svega svaki korak obavezno spremite kako bi imali prethodno stanje.

N:Sada možemo pogledati osnovne skupine kistova

N:Postoji cijela paleta kistova: Calligraphic, Art, Pattern i Bristle. Prvo ćemo koristiti kaligrafski kist (Calligraphic brush) da bismo ukrasili vagon s polukrugovima u boji.Inače kaligrafija se može interpretirati kao lijepo pisanje s određenim konturama i debljinom linija. S desne strane u odaberemo Control tool i u izborniku isključimo sve druge kistove osim Calligraphic brusha

Zašto isključujemo ostale vrste kistova? Zato da bi se lakše usredotočili na pojedinačan kist iz palete kistova i upoznali sve njegove mogućnosti.

N: Kaligrafski kistovi (Calligraphic brushes) služe za crtanje krivulje pod vrlo izvučenim kutom kao kad bi radili s perom ili onim plosnatim markerom. Tako na određenim dijelovima krivulja može biti široka, a na dijelovima uska što stvara vrlo ugodan dojam.

(Sada provodim korake navedene u dokumentu "Upute za rad u Adobe Illustratoru"uz neke popratne komentare)

N: Kada ste primijenili sve ove korake, trebali biste dobiti sljedeće:

(Prikažem rješenje)



N: Jeste li uspjeli? Ako netko nije uspio napraviti molim da podigne ruku?

(U slučaju da su svi gotovi idem dalje u suprotnom odlazim do učenika i pomažem mu u realizaciji koraka vježbe)

N:Kistove je moguće i uređivati da bi se dobio ljepši crtež odnosno da bi napravljena krivulja izgledala vjerodostojnije. Za realizaciju toga otvaramo Calligraphic Brush Option dialog box.

## (Provodim korake kao u uputama)

N: Jeste li uspjeli? Ako netko nije uspio napraviti molim da podigne ruku?

(U slučaju da su svi gotovi idem dalje u suprotnom odlazim do učenika i pomažem mu u realizaciji koraka vježbe)

N:Uspješnim određivanjem opcija za kaligrafske kistove, dobili smo: (Prikažem rješenje)



N: O ovim koracima bio je korišten kaligrafski kist. Zbog činjenice da kaligrafski kist ne može zadovoljiti baš sve vrste funkcionalnosti, prelazimo na umjetnički kist (Art brush). Umjetnički kistovi nam daju mogućnosti stvaranja raznih tekstura. Prvo se pozicioniramo oko medvjedove glave pomoću Zoom tool i označimo je. Učinit ćemo da medvjed bude nalik pravome medvjedu pa dodajmo krzno po dijelovima njegova tijela. Takvu nakanu ostvarit ćemo pomoću "dlakave" teksture koju nam nudi umjetnički kist (Art brush). No prije toga u izborniku kistova isključimo sve kistove osim Art brusha kao i kod prethodnog korištenja kaligrafskog kista iz istih razloga da se lakše usredotočimo na mogućnosti koje nam daje umjetnički kist.

(Provodim korake prema "Upute za rad u Adobe Illustatoru")



N: Jeste li uspjeli? Ako netko nije uspio napraviti molim da podigne ruku?

(U slučaju da su svi gotovi idem dalje u suprotnom odlazim do učenika i pomažem mu u realizaciji koraka vježbe)

#### Zadatak(4 minute)

N: Sad vas molim da osim krzna učinite i uha na ovom medvjediću čupavim upotrebom istih funkcionalnosti umjetničkog kista.

(Ostavim ih 4 minute da sami naprave. Prošećem razredom i kada vidim da su gotovi ili nisu uspjeli to napraviti, sam prezentiram rješenje, ali prije toga pitam one koji su to uspjeli napraviti kako su to napravili da malo pojasne). N: Kao i kod kaligrafskih kistova i umjetnički kistovi imaju odgovarajuće mogućnosti uređivanja. Umjetnički kist ima mogućnost kreiranja vlastite vrste umjetničkog kista. Prvo u glavnoj alatnoj traci na vrhu radnog prostora odaberite klikom miša View>Fit Artboard in Window kako biste dobili bolji pregled nad cijelim radnim prostorom. Sada dođite do drugog crteža na desnoj strani Document Windowa koristeći desnu strelicu klizača te pozicionirajte pokazivač miša na crne zvjezdice i jednim klikom na odabranu poziciju označite te crne zvjezdice. Želimo na lokomotivi napraviti vijenac oko znaka RR.

(Provodim korake prema "Upute za rad U Adobe Illustatoru")



N: Jeste li uspjeli? Ako netko nije uspio napraviti molim da podigne ruku?

(U slučaju da su svi gotovi idem dalje u suprotnom odlazim do učenika i pomažem mu u realizaciji koraka vježbe)

N: Sljedeće ćemo se baviti s čekinjastim ili četkastim kistovima (Bristle brushom). Istim principom kao u prethodna dva slučaja u izborniku kistova isključite sve kistove osim četkastog kista (Bristle brusha). Četkasti kist omogućava efektno crtanje. To znači da se naglašavaju određeni efekti kao gorenje vatre primjerice. Četkasti kist daje prirodan ton crtežu i osjećaj da je crtež stvaran.

N: No prije konkretizacije zadataka s četkastim kistom, moramo postaviti određene parametre u Brush Option dialog box.

(Provodim korake prema "Upute za rad U Adobe Illustatoru")

Name: Stort				1	OK
Shape: Ja Flat O	roe			-	Cancel
Brush Options -				-	Previou
					Carl ( a Date
			-		
Size			3		
Sile			3 1010		
Bristie Length:	Short	Long	178	96	
Bristle Density	Low	High	84	96	
Bristle Thickness	Fine	Coarse	74	*	
Paint Opacity:	Translucent	Opaque	90	*	

N: Jesu svi uspjeli posložiti parametre? Ako niste sve parametre postavili kako treba, molim pogledajte u upute pod točkom 4.1. Mijenjanje opcija četkastih kistova.

N:Sada možemo oslikati vatru ispred zmaja na vagonu. Prije si upotrebom Zoom tool povećajte mjesto vatre te ju označite klikom miša.

(Provodim korake prema ,, Upute za rad U Adobe Illustatoru")

N:Ako ste pravilno radili svi biste trebali dobiti nešto poput ovoga:

(Demonstriram crtež)



N: Jeste li uspjeli? Ako netko nije uspio napraviti molim da podigne ruku?

(U slučaju da su svi gotovi idem dalje u suprotnom odlazim do učenika i pomažem mu u realizaciji koraka vježbe)

N: Na početku smo govorili da je Adobe Illustrator aplikacija koja radi s vektorskim zapisom.

N:Može li netko ponoviti što to znači vektorski?

U: Vektorski zapis računalo pamti kao matematičke funkcije koje opisuju krivulje.

N:Tako je. Sad ćemo pokazati taj vektorski zapis. Na alatnoj traci na vrhu radnog prostora kliknite na View>Outline. Primijetite linije, krivulje, ponajviše obratite pažnju na plamen i mjesto po kojem ste povlačili linije uz pomoć četkastog kista.

N:Da bi vratili crtež u normalu na alatnoj traci pri vrhu radnog prostora kliknite na View>Prewiev

#### Zadatak (trajanje 5 minuta)

N: Promijenite boju vagona po želji uz primjenu četkastog kista uvažavajući sve korake koje smo primijenili na vatri. Pazite kako i što označavate te gdje se pozicionirate na kojem sloju.

(Ostavim ih 5 minute da sami naprave. Prošećem razredom i kada vidim da su gotovi ili nisu uspjeli to napraviti, sam prezentiram rješenje, ali prije toga pitam one koji su to uspjeli napraviti kako su to napravili da malo pojasne). N: Sada je vrijeme za odmor i zvonilo je. Možete na 5 minuta izaći i malo se osvježiti.

#### Glavni dio sata 2. Sat (40 minuta)

#### (Pričekam da se svi vrate i toleriram 1-2 minute kašnjenja)

N: Nastavljamo raditi dalje s preostalim vrstama kistova. Još su nam preostali Pattern brushes i Blob brushes. Upotrijebit ćemo njihove mogućnosti u dovršavanju crteža na vrlo sličan način odabirom gotov jednakih opcija kao kod prethodnih vrsta kistova umjetnički kist, kaligrafski kist i četkasti kist).

N: Kako prevodimo riječ Pattern, a kako Blob naravno u kontekstu kistova?

U:Pattern prevodimo kao uzorak, a Blob kao grudičasto.

N: Prema tome Pattern brushes su kistovi uzoraka kojima crtamo na temelju odabranih gotovih oblika odnosno uzoraka, dok s Blob brushes ili grudičastim kistovima dobivamo crtamo grudičaste strukture, nešto nalik nagnječenoj masi. Kada želimo koristiti kist uzoraka odaberemo uzoraka, povlačeći taj kist, on ostavlja identične tragove s tim uzorkom na određenoj udaljenosti. Najprije ćemo kist uzoraka koristiti za crtanje pruge ispod vlaka.Najprije na alatnoj traci na vrhu radnog prostora odaberemo View>Fit Artboard In Window kako bismo dobili bolji pregled i isključimo u Brush panel izborniku sve kistove osim kistove uzoraka (Pattern Brushes).

(Provodim korake prema "Upute za rad u Adobe Illustratoru").

N: Idem li prebrzo ili vam je ovaj tempo sasvim dobar?

(Ako je dobar nastavim s radom, ako idem prebrzo malo usporim i prilagodim se učenicima)



N: Kako vam se čini rad s ovim kistom? Je li s njim lakše raditi nego s ostalim kistovima?

U: Raditi s ovim kistom je puno jednostavnije jer ima mali broj koraka za odabir samog kista i upotrebu tog istog kista.

N:Tako je. Kistovi uzoraka se upotrebljavaju po jednostavnome principu. Jednostavno dodamo označeni i odabrani uzorak direktno ga primjenjujemo u crtanju.

N:Sada ćemo primijeniti sličan način crtanja po okviru slike.Ovaj dio je specifičan po tome što je moguće dva uzorka spojiti u jedan kist uzoraka. Kreiramo novi kist putem odabiranja željenog uzorka koji se nalazi u drugom artbordu s desne strane radnog prostora do kojeg možete doći klikom na lijevu tipku klizača. Odaberemo uzorak cvjetića pa nakon toga ovaj uzorak narančastog kružića i stopimo ih u jedan kist uzorka.

Name:	Border				C	OK
Scale:	70%	1.00%	Fa	ed	- 6	Cancel
Spicing	120%				5	Preyew
1	<b>●</b> 行		NP	Filp Elip Along El Filp Agros	14 	
None	-		-	Fit		
Correr	14			C Stretch to	tit.	
5408				Appcoxim	to fit ste path	
				- Colorization		
				Method N	one	*
				Rev. Pater	-	~

(Provodim korake prema "Upute za rad u Adobe Illustratoru").

N: Jeste li uspjeli? Ako netko nije uspio napraviti molim da podigne ruku?

(U slučaju da su svi gotovi idem dalje u suprotnom odlazim do učenika i pomažem mu u realizaciji koraka vježbe)



### Zadatak (trajanje 10 minuta)

N: Sami ćete sada uz pomoć uputa isprobati funkcionalnost mijenjanja boja koristeći Tint colorization i Hue Shift colorization. To vam se nalazi pod točkom 6., točnije 6.1. i 6.2. u uputama. Imate točno 10 minuta za izradu rješenja. Možete krenuti.

(Povremeno svraćam do učenika i pomažem im u izvedbi rješenja. Nakon 10 minuta prelazim na daljnje demonstracije).

N: Prelazimo na rad s posljednjim kistom takozvanim grudičastim kistom (Blob Bruh tool). Grudičasti kist služi za spajanje s drugim oblicima ili za stvaranje ispuna gdje je potrebna struktura u obliku oblaka u obliku nagomilane mase.

To ćemo upotrijebiti za izradu dima iznad naše lokomotive. Dakle prvo ćemo koristiti prvu funkcionalnost grudičastog kista, mogućnost stvaranje dima odnosno za ispunu.

N: Najprije u alatnoj traci na vrhu radnog prostora kliknemo na View> Fit Artboard In Window. Na izborniku sa slojevima onemogućimo vidljivost (visibility) nad Text layerom, Background Layerom i Smoke Layerom radi lakše izvedbe i rada s grudičastim kistom.

(Sada provodim korake navedene u dokumentu "Upute za rad u Adobe Illustratoru"uz neke popratne komentare)



N: Jeste li uspjeli napraviti ? Ako netko nije uspio napraviti molim da podigne ruku?

(U slučaju da su svi gotovi idem dalje u suprotnom odlazim do učenika i pomažem mu u realizaciji koraka vježbe)

N: Prelazimo na drugu funkcionalnost grudičastog kista, a to je spajanje Blob Brush tool odnosno grudičastog kista s nekim elementom (u našem primjeru radimo povezivanje s elipsom).

(Demonstriram po koracima iz dokumenta "Upute za rad u Adobe Illustratoru")

N: Jeste li uspjeli napraviti ? Ako netko nije uspio napraviti molim da podigne ruku?

(U slučaju da su svi gotovi idem dalje u suprotnom odlazim do učenika i pomažem mu u realizaciji koraka vježbe)

Mode: Multiply +	OK
Quacity: 15 + N	Cancel
Coffset 💿 3 pe	Printer
Coffset 2 3 pt	
Bkr 2 pt +	
Color Coarkness: 100 N	

## Zadatak (trajanje 10 minuta)

Iskušajte sve do sad naučene metode i mogućnosti rada s brushevima nad istom početnom slikom u datoteci Lession11 pod nazivom L11start\_1.ai s tim da uključite i neke izmjene:

- Primijenite Bristle brush nad oblakom kao što ste ga primjenili nad vatrom za postizanje istog efekta realističnosti
- Primjenite Bristle brush nad zmajem uistom svojstvu kao nad vatrom
- Koristite Art brush i dodajte piletu (vozač vlaka) po žutom licu paperjes lično kao na medvijedu dlaku samo u drugoj boji
- Učitajte u Art brush uzorak trave isto kao što smo napravili s Pattern brushom za tračnice, iscrtajte travu uz vlak

(Ovo mi je pričuvni zadatak ukoliko dođe do situacije da ostane vremena više od očekivanog u glavnom djelu drugog sata).

### Završni dio sata (5 minuta)

(Postavljam par pitanja za ponavljanje gradiva)

N:Koje vrste Brusheva smo danas isprobali?

U: Calligraphic, Art, Pattern i Bristle.

N:Po kojoj specifičnosti je poznat kist uzoraka osim što se kreira odabirom odgovarajućeg uzorka.

U: U jedan kist uzoraka, moguće je objediniti dva pojedinačna uzorka.

N: Za što smo koristili opciju View> Fit Artboard In Window?

U:Da bismo se vratili u početni položaj po pitanju perspektive te da bismo imali bolji pregled nad cijelom radnom površinom.

N: Sad će zvoniti. Možete se spremiti i isključiti računala. Zahvaljujem na suradnji.

## Kraj pripreme

## **1.2.** Primjer pisane pripreme 2. – javno predavanje

Sljedeća priprema pisana je za izvođenje nastavnog sata na **Elektrostrojarskoj školi Varaždin** za 2. razred usmjerenja **Tehničar za računalstvo.** Obrađivalo se područje pogramiranja u programskom jeziku C++. Za razliku od prethodne pripreme, ova priprema ima makroplan i mikroplan za izvođenje nastavnog sata. Također u pripremi su navedene i reference prezentacije predavanja čiji sadržaj ovdje nije prikazan zbog opsežnosti.

Početak pripreme:

Škola: Elektrostrojarska škola Varaždin Razred: 2.b Nastavni predmet: Računalstvo i programiranje Nastavna cjelina: Funkcije u C++ Nastavna jedinica: Vrste funkcija Tip sata: obrada novog gradiva i uvježbavanje Blok sat: da

## PLAN VOĐENJA (ORGANIZACIJE) NASTAVNOG PROCESA

## Cilj obrade nastavne jedinice

Upoznavanje učenika s vrstama funkcija u jeziku C++ i uvježbavanje.

## Zadatci koje treba ostvariti da bi se postigao cilj

Materijalni zadatci:

- ponoviti što su funkcije i deklaracije i definicije funkcija
- razlikovati funkcije s povratnom i bez povratne vrijednosti
- prepoznati potrebu za umetnutim i preopterećenim funkcijama
- prepoznati korisnost rekurzivnih funkcija i primijeniti ih
- prepoznati korisnost predložaka funkcija i primijeniti ih
- nabrojati najčešće korištene standardne funkcije, razumjeti svrhu i potrebu postojanja standarda, naučiti prepoznati funkcije preuzete iz C jezika
- razlikovati biblioteke programskog koda od objektnog, shvatiti potrebu za postojanjem biblioteka, napisati i uključiti programski kod i objektni kod u glavni program

Funkcionalni zadatci:

- primijeniti prikazane algoritme
- razvijati sposobnost apstraktnog mišljenja, odnosno programskog načina razmišljanja i zapisivanja
- razvijati sposobnost analitičkog pristupa problemu

## Odgojni zadatci:

- formirati pozitivne stavove prema radu
- razvijati smisao za urednost, upornost i ustrajnost u radu

## Nastavne metode

- Metoda razgovora
- Metoda usmenog izlaganja
- Metoda demonstracije
- Metoda praktičnih aktivnosti

## **Oblici rada**

- Frontalni rad
- Individualni rad

## Nastavna sredstva i pomagala

- Računalo
- Projektor
- PowerPoint prezentacija
- Visual C++ 2008 Express Edition

## Izvori znanja

- Programiranje II: materijali s predavanja i laboratorijskih vježbi, Dr. sc. Danijel Radošević
- Programiranje I: materijali s predavanja, auditornih i laboratorijskih vježbi prof. dr. sc. Alen Lovrenčić
- Šribar J., Motik B.; Demistificirani C++, drugo izdanje, Element, Zagreb 2006.

## ORGANIZACIJA NASTAVNOG RADA – ARTIKULACIJA METODIČKE JEDINICE (makroplan)

REDNI BROJ	FAZE RADA I SADRŽAJ	METODIČKO OBLIKOVANJE	VRIJEME U MINUTAMA
1.	<ul> <li>UVODNI DIO <ul> <li>Pozdravljanje i upoznavanje s učenicima (kratko predstavljanje da ću ja danas održati predavanje)</li> <li>Dogovor o načinu rada (javljanje dizanjem ruku)</li> <li>Predstavljanje sadržaja nastavne jedinice (popis aktivnosti koje će se odraditi)</li> </ul> </li> </ul>	<ul> <li>Usmeno izlaganje</li> <li>Razgovor</li> <li>Frontalni rad</li> </ul>	5'
2. 2.1.	SREDIŠNJI DIO Ponavljanje - Ponavljanje temeljnih pojmova iz prethodne nastavne jedinice kao motivacija i uvod u obradu novog sadržaja	<ul> <li>Usmeno izlaganje</li> <li>Razgovor</li> <li>Demonstracija</li> <li>Praktična aktivnost</li> <li>Frontalni rad</li> </ul>	35'
2.2.	<ul> <li>Obrada novog sadržaja 1.</li> <li>Funkcije s i bez povratne vrijednosti (teorija i praktični primjeri)</li> <li>Umetnute funkcije (teorija i praktični primjer)</li> <li>Preopterećene funkcije (teorija i praktični primjer)</li> </ul>		

	<ul> <li>Rekurzivne funkcije (teorija i praktični primjer)</li> <li>Predlošci funkcija (teorija i praktični primjer)</li> </ul>		
3.	MALI ODMOR		
4.	SREDIŠNJI DIO	- Razgovor	40'
4.1.	<ul> <li>Obrada novog sadržaja 2.</li> <li>Standardne funkcije (teorija i praktični primjer)</li> <li>Biblioteke (teorija i praktični</li> </ul>	<ul> <li>Demonstracija</li> <li>Praktična aktivnost</li> <li>Frontalni rad</li> </ul>	
	primjeri kroz pisanje i uključivanje izvornog i objektnog koda)	- Individualni rad	
4.2.	Vježba		
	<ul> <li>Zadatak 1 (individualni rad – zbroj prvih n prirodnih brojeva)</li> </ul>		
	<ul> <li>Zadatak 2 (individualni rad – zbroj prvih n neparnih prirodnih brojeva koristeći rekurziju)</li> </ul>		
	<ul> <li>Zadatak 3 (individualni rad – tornjevi Hanoja)</li> </ul>		
	ZAVRŠNI DIO	- Razgovor	5'
	- Ponavljanje	- Frontalni rad	
	- Najavljivanje zvona		
5.	<ul> <li>Napomena da učenici pospreme radna mjesta</li> </ul>		
	- Pozdravljanje		

## TIJEK IZVOĐENJA NASTAVE – NASTAVNI RAD (mikroplan)

## 1. Uvodni dio (5 minuta)

U ovom dijelu sata se predstavljam učenicima, dogovaram način rada i učenicima predstavljam popis aktivnosti koji ćemo odraditi u nastavnoj jedinici.

N: Dobar dan!

U: Dobar dan!

N: Ja sam X. X; student Fakulteta organizacije i informatike i danas ću vam održati blok sat nastave iz informatike. Danas ću ja biti vaš nastavnik.

Najavljujem današnju temu. (.ppt p1)

Na prezentaciji pokazujem aktivnosti koje ćemo odraditi kroz blok sat. (.ppt p2)

Učenike upoznajem s pravilima rada.

N: U slučaju da imate pitanje ili želite odgovoriti na moje pitanje molim vas podignite ruku. Također vas molim da računala ne koristite u druge svrhe, osim za potrebe zadataka koje ćemo danas izraditi.

## 2. Središnji dio 1 (35 minuta)

U ovom dijelu se izmjenjuje ponavljanje, obrada novog sadržaja kroz razgovor i usmeno izlaganje, rješavanje primjera zadataka samostalno bez nastavnikove pomoći i kroz demonstracije.

## **2.1. Ponavljanje** (5 – 7 minuta)

S učenicima u obliku pitanja kroz razgovor popraćen prezentacijom ponavljam što su radili u prethodnoj nastavnoj jedinici. Ovo ima dvostruku svrhu: radi se logički slijed između nastavnih jedinica; učenici ponavljaju što su radili, ali istodobno daju i povratnu informaciju nastavniku kakva im je razina znanja kao bi nastavnih znao prilagoditi tempo obrade novog gradiva; istodobno, eventualni nedostatak potrebnog predznanja se kroz ponavljanje nadoknađuje.

Pitanja za ponavljanje kao uvod u obradu novog sadržaja: (.ppt p3, 4, 5)

N: Što je to funkcija u C++-u?

U: Potprogram – izdvojena logička cjelina unutar programa (sa svojim nazivom i tipom, parametrima, lokalnim podatcima i povratnom vrijednosti).

N: Zašto se koriste funkcije u C++-u?

U: Zato da bi se izbjeglo ponavljanje pisanja koda, da bi se izbjegla glomaznost i postigla modularnost koda, da bi kod postao čitljiviji i razumljiviji, da bi se lakše otkrile i ispravile pogrješke.

N: Što je to deklaracija funkcije?

U: Prototip funkcije ili glava/zaglavlje funkcije kojim se određuje tip, naziv i potpis funkcije.

N: Što je to potpis funkcije?

U: Broj i tip formalnih argumenata (parametara) funkcije.

N: Što je to definicija funkcije?

U: Glava funkcije + tijelo funkcije

- N: Od čega se sastoji tijelo funkcije?
- U: Od lokalnih podataka, instrukcija i povratne vrijednosti.
- N: Kakvi su to lokalni podatci?
- U: Podatci koji su dostupno samo unutar funkcije.

N: Koja je razlika između formalnih i stvarnih argumenata?

U: Formalni argumenti se navode u zaglavlju (deklaraciji) funkcije i oni predstavljaju lokalne podatke funkcije koji se alociraju na stogu, a stvarni argumenti se navode kod poziva funkcije prilikom čega formalni argumenti poprimaju vrijednosti odgovarajućih stvarnih argumenta.

## 2.2. Obrada novog sadržaja 1. (28 – 30 minuta)

Svaki korak u obradi novog sadržaja uključuje i kod s praktičnim primjerom.

Svaki korak ima svoj jedinstveni naziv i zapravo predstavlja vrstu funkcije.

## Funkcije s i bez povratne vrijednosti.

Teoretski dio: (.ppt p6)

Funkcija u jeziku C++ može imati više izlaza čime se narušava pravilo pravilnog programa!

Tip funkcije određuje tip podatka koji funkcija vraća pozivajućem programu.

Naredbom return se prekida izvođenje funkcije te se vrijednost vraća pozivajućem programu.

Ako je rezultat izraza naredbe *return* različitog tipa od tipa funkcije, rezultat se pravilima pretvorbe svodi na tip funkcije.

Funkcija koja ne vraća vrijednost umjesto tipa ima oznaku void.

Ona ne sadrži povratnu vrijednost iza return ili uopće ne sadrži *return* (povratak je na kraju funkcije)

Praktični primjeri (.ppt p7, 8)

## Umetnute funkcije

Objašnjavam da umetnute funkcije predstavljaju zamjenu poziva funkcije samim kodom funkcije.

Putem ključne riječi *inline* daje se naputak prevoditelju da pokuša svaki poziv funkcije nadomjestiti kodom funkcije.

Naglašavam da je to samo naputak, a ne naredba prevoditelju tako da će uspjeh prevođenja ovisiti o samom prevoditelju i složenosti same funkcije. (.ppt p9)

Učenici zatim rade modifikaciju prethodnog rješenja zadatka koristeći ključnu riječ *inline*. (.ppt p10)

## Preopterećenje funkcije

Objašnjavam kroz primjer da se pod tim nazivom podrazumijeva korištenje istog naziva za različite implementacije funkcije, a kako bi se to ostvarilo preopterećene funkcije se moraju razlikovati po potpisu funkcije. (.ppt p11, 12)

Zatim postavljam pitanje kao bih dobio uvid jesu li učenici zapamtili što je to potpis funkcije.

N: Što je to potpis funkcije?

U: Broj i tip argumenata funkcije.

Kroz praktični primjer učenici se upoznaju s preopterećenjem funkcija. (.ppt p12)

## Rekurzivne funkcije

N: Što je to rekurzija u C++-u?

U: Svojstvo potprograma da može pozivati sam sebe.

Da bi se to svojstvo postiglo potrebni su: sidreni izraz (uvjet) koji omogućava izlaz iz rekurzivne funkcije (u protivnom funkcija radi beskonačni poziv same sebe) i rekurzivna formula ili postupak koji trebaju pojednostaviti zadatak, odnosno, približiti ga sidrenom izrazu. (.ppt p13)

Napominjem da kod primjene rekurzija treba biti oprezan s obzirom da se lokalne varijable i adrese s kojih se obavlja poziv funkcije spremaju na stog tako da se prevelikim uzastopnim pozivom funkcije stog lako može prepuniti što će onemogućiti daljnje izvođenje programa. (.ppt p14, 15)

Kroz praktični primjer učenici isprobavaju rekurzivni poziv funkcije.

## Predlošci funkcija

Predlošci funkcija predstavljaju parametrizirane funkcije kod kojih se tip funkcije i tip parametara (argumenata) mogu mijenjati po potrebi. (.ppt p16)

Kao bih motivirao učenike (s obzirom da se već bliži kraj sata) prije praktičnog primjera dajem primjer za motivaciju. (.ppt p17)

Kroz praktični primjer učenici se upoznaju s prednošću korištenja predloška funkcija s obzirom da za razliku od motivirajućeg primjera koji se sastoji od devet linija koda sada to mogu riješiti u četiri linije koda (.ppt p18)

Najavljujem zvono.

## 3. Mali odmor

Pauza između blok sata je predviđena u trajanju 5 minuta.

- 1. sat (07:20 8:00)
- 2. sat (08:05 08:50)

## 4. Središnji dio 2 (40 minuta)

Prije obrade novog sadržaja radim uvod u obliku podsjećanja učenika što smo do sada odradili (funkcije s i bez povratne vrijednosti, umetnute funkcije, preopterećene funkcije, rekurzivne funkcije i predloške (paremetrizirane funkcije)) i što ćemo još napraviti (standardne funkcije i moduli s programskim kodom i objektnim kodom)

## 4.1. Obrada novog sadržaja 2. (20 minuta)

## Standardne funkcije

C++ nema ugrađene funkcije u sam jezik, ali zato postoji velik broj funkcija koji se isporučuje u bibliotekama zajedno s prevoditeljem.

208 standardnih funkcija je preuzeto iz jezika C, a te je funkcije moguće prepoznati po ključnom slovu c ispred naziva funkcije.

Navodim da je standardne funkcije potrebno uključiti pretprocesorskom naredbom #include, a zatim učenike pitam koje su standardne funkcije do sada koristili i postavljam podpitanje za što su to koristili.

N: Koje ste standardne funkcije do sada koristili?

U: iostream - osnovni ulazno izlazni C++ tokovi; cmath - osnovne matematičke funkcije; ...

Kroz praktični primjer učenici se upoznaju s funkcijom ctime (.ppt p21)

#### Biblioteke

Dijelovi programa koji se uključuju u matičnu programsku datoteku ili se iz nje pozivaju.

U C i C++ biblioteke se uključuju pomoću pretprocesorske instrukcije #include.

Učenicima govorim da je moguće uključiti programski kod ili (pomoću datoteka zaglavlja header.h) objektni kod.

Kroz praktični primjer učenici isprobavaju uključivanje biblioteka s programskim kodom i objektnim kodom. (.ppt p24, 25, 26, 27)

## 4.2. Vježba (20 minuta)

Nakon uvoda u zadatak učenici rješavaju zadatke samostalno, a nastavnik radi provjeru napretka rješenja, nakon čega daje prikaz cjelovitog rješenja svakog od zadataka. (.ppt p28 – 44)

U prva dva zadatka učenicima je dan tekstualni opis zadatka što trebaju napraviti (.ppt p29, 31), a u trećem zadatku je osim teksta zadatka (.ppt 33) dana i vizualizacija rješenja korak po korak (.ppt 34 – 41), verbalizacija rješenja (.ppt 42) i pseudokod rješenja (.ppt 43)

U ovom dijelu dajem učenicima mogućnost, ako netko od učenika to želi, napraviti demonstraciju rješenja zadatka na nastavnikovom računalu ostalim učenicima, u suprotnom dajem prikaz koda rješenja.

## 5. Završni dio (5 minuta)

Ponavljanje što smo danas obradili. Postavljam pitanje koje smo vrste funkcija danas obradili, a za svaku navedenu vrstu funkcije postavljam potpitanje na što se ta funkcija odnosi.

N: Koje smo sve vrste funkcija danas obradili?

U: Funkcije s i bez povratne vrijednosti, umetnute funkcije, preopterećene funkcije, rekurzivne funkcije, predlošci (parametrizirane, prototipizirane funkcije), standardne funkcije, biblioteke (moduli, datoteke zaglavlja)

N: Kako razlikovati funkciju bez povratne vrijednosti od funkcije s povratnom vrijednosti?

U: Funkcija koja ne vraća vrijednost umjesto tipa ima oznaku void, ne sadrži povratnu vrijednost iza return ili uopće ne sadrži return.

N: Kako prepoznati umetnutu funkciju?

U: Preko ključne riječi inline ispred tipa funkcije.

N: Što je to preopterećenje funkcije?

U: Korištenje istog naziva za različite funkcije.

N: Što je to rekurzivna funkcija?

U: Funkcija koja poziva samu sebe.

N: Za što su nam korisni predlošci funkcija?

U: Omogućavaju nam da se jednom definirani kod prevede više puta za različite tipove argumenata. Oslobađa nas opetovanog pisanja identičnih funkcija koje se razlikuju samo po tipu.

N: Čemu služe standardne funkcije?

U: Olakšavaju nam pisanje programa bez bojazni po prenosivost koda.

Napomena učenicima da pospreme radna mjesta.

N: Molim vas pospremite radna mjesta prije nego što izađete iz učionice

Pozdrav učenicima.

N: Hvala vam na suradnji.

Kraj pripreme

## **1.3.** Primjer pisane pripreme 3. – javno predavanje

Sljedeća priprema je iznimno detaljna u odnosu na prethodne dvije. Nastavni sat je održan na Elektrostrojarskoj školi Varaždin u drugom razredu na nastavnom predmetu Izrada web stranica.

Početak pripreme:

## Priprema za izvođenje nastavnog sata

Nastavnik		X.X.	Nadnevak	15.05.2014.		
Škola		Elektrostrojarska škola Varaždin	Razred	2. razred		
Nastavni predmet		Izrada web stranica				
Nastavna cjelina		CSS3				
Nastavna jedinica		CSS3/JS animacije				
Cilj nastavne jedinice		Utvrđivanje znanja iz CSS-a i JS-a s kojim se mogu izraditi animacije na web stranicama				
Tip nastavnog sata		Ponavljanje i vježbanje gradiva				
Zadaci	Obrazovni	Ponoviti, razumjeti i moći primijeniti HTML – osnovnih elemenata i pripadajućih atributa Ponoviti, razumjeti i moći primijeniti CSS3 – osnovnih svojstava i svojstava potrebnih za animiranje HTML elemenata Ponoviti, razumjeti i moći primijeniti JS – varijable, funkcije, događaji				
	Funkcionalni	Razvijati sposobnost samostalnog rada s HTML-om, CSS- om i JS-om te razviti sposobnost prepoznavanja i ispravljanja grešaka u kodu				
	Odgojni	Poticati pozitivne stavove prema radu Poticati samostalnost Poticati osjećaj za preglednost, red, urednost i sistematičnost				

Nastavne metode	Metoda razgovora Metoda usmenog izlaganja Metoda demonstracije Metoda praktičnih aktivnosti
Socijalni oblici rada	Frontalna nastava Individualni rad / Rad u paru (u slučaju manjka računala)
Nastavna sredstva i pomagala	Računalo Projektor
Izvori znanja	http://www.w3schools.com/

# Organizacija nastavnog rada – artikulacija metodičke jedinice (makroplan)

Koraci nastavnog procesa i trajanje u minutama		Sadržaj koraka	Socijalni oblici rada i nastavne metode
Uvodni dio	5	Pozdrav i predstavljanje Upoznavanje s pravilima i načinom rada Provjeravanje tko je odsutan Ponavljanje predznanja iz HTML-a Upoznavanje učenika sa sadržajem današnjeg nastavnog sata	Metoda razgovora Metoda usmenog izlaganja Frontalna nastava
Glavni dio	70	<ul> <li>1. zadatak – CSS korištenje svojstava height, width, background, position, bottom, left (sa pripadajućim vrijednostima); CSS3 animacija pomoću transition i transform (sa pripadajućim vrijednostima); HTML korištenje elemenata html, head, title, style, body, div (sa pripadajućim atributima)</li> <li>2. zadatak – CSS korištenje svojstava background, color, width, height, line- height, text-align, position (sa pripadajućim vrijednostima); CSS3 animacija pomoću transition (sa pripadajućim vrijednostima); HTML korištenje elemenata html, head, title, style, script, body (sa pripadajućim atributima); JS korištenje globalnih i lokalnih varijabli i funkcija</li> <li>3. zadatak – CSS korištenje svojstava background, border, padding, width, margin, height, overflow (sa pripadajućim vrijednostima); CSS3 animacija pomoću transition (sa pripadajućim vrijednostima); HTML korištenje elemenata html, head, title, style, body, div, h2, h3, a, p, br (sa pripadajućim atributima)</li> </ul>	Metoda razgovora Metoda usmenog izlaganja Metoda demonstracije Metoda praktičnih aktivnosti Frontalna nastava Individualni rad / Rad u paru
Završni dio	15	- 4. zadatak – samostalni zadatak (pronalaženje i ispravljanje grešaka u kodu): lampa_greska.html (greške na linijama: 11, 12, 17, 22, 25, 27, 37, 50, 56, 65, 69, 70, 74, 81, 93, 96, 103, 119, 124, 127, 132, 138, 140, 143, 148)	Metoda razgovora Metoda usmenog izlaganja Metoda praktičnih aktivnosti Frontalna nastava Individualni rad / Rad u paru

## Tijek izvođenja nastave – nastavni rad (mikroplan)

## Uvodni dio

U uvodnom dijelu sata ću se predstaviti učenicima, iznijet ću im pravila i način izvršavanja sata, upoznat ću ih sa sadržajem današnjeg sata te ću utvrditi predznanje koje imaju.

N: Dobro jutro!

#### U: Dobro jutro!

N: Moje ime je X.X. i danas ću biti vaša nastavnica. Prije nego što krenemo sa radom, jedno pravilo – ako netko ima pitanje ili želi odgovoriti na moje pitanje, molim vas da prvo digne ruku. Kad vidim ruku u zraku, prozvat ću vas i dati vam riječ. Ako slučajno ne primijetim odmah ruku, ili sam usred rečenice, dat ću vam znak da sam vas vidjela i prozvat ću vas čim završim.

N: Molila bih redare da se ustanu i kažu mi koga nema na satu.

#### U: [govore tko nije prisutan]

**N:** Danas ćemo ponavljati CSS3, JS i animacije. Riješit ćemo nekoliko zadataka zajedno, a na kraju ćete dobiti samostalni zadatak. Ukoliko imate pitanja, pitajte. Na sljedećem linku možete preuzeti materijale potrebne za izvršavanje današnjih zadataka. Ako netko ima problema sa linkom ili ne može preuzeti na taj način, neka digne ruku pa ću mu poslati USB.

N: Dok preuzimate materijale ćemo kratko ponoviti osnove HTML-a, CSS-a i JS-a.

N: Koji su osnovni elementi u HTML dokumentu?

U: Osnovni elementi u HTML dokumentu su <html>, <head>, <body>.

N: Tako je – da bismo uopće krenuli pisati bilo kakav kod, trebamo prvo otvoriti i zatvoriti <html> i unutar njega upisati <head> i <body>.

N: Kako izgleda otvoreni, a kako zatvoreni tag?

**U**: U tag-ovima koristimo izlomljene zagrade unutar kojih pišemo naziv elementa. Otvoreni tag se sastoji samo od zagrada i naziva (<...>), dok zatvoreni tag ima i kosu crtu prije naziva elementa koji se zatvara (</...>).

N: Tako je – koristimo izlomljene zagrade, ili, drugačije rečeno, koristimo znakove za manje i veće i unutar njih naziv elementa. Kad zatvaramo tag, koristimo kosu crtu. Važno je da naziv elementa koji smo otvorili bude jednak nazivu elementa koji zatvaramo.

N: Kad koristimo CSS, koji element koristimo, koji atribut moramo navesti i unutar kojeg elementa ga najčešće pišemo?

U: Koristimo <style>, u kojeg upisujemo atribut type sa vrijednosti *text/css*. <style> element upisujemo unutar <head> elementa.

N: Tako je – unutar <head> elementa upisujemo <style> element i njegov atribut type sa vrijednosti *text/css*.

N: Koliko načina postoji za ubacivanje CSS-a u HTML dokument i koji su?

U: Postoji tri načina za ubacivanje CSS-a: interni, eksterni i inline.

N: Tako je – tri su načina za ubacivanje CSS-a: interni, eksterni i inline. Kad ubacujemo CSS, koji god način koristili, ključna riječ nam je style. Interni je koristeći <style> element, eksterni koristeći <link rel="stylesheet" type="text/css" href="\*.css" /> gdje CSS pišemo u zasebnom dokumentu, te inline kad upisujemo CSS za svaki HTML zasebno koristeći HTML atribut style.

N: Kad koristimo JS, koji element koristimo, koji atribut moramo navesti i unutar kojeg elementa ga najčešće pišemo?

U: Koristimo <script>, u kojeg upisujemo atribut type sa vrijednosti *text/javascript*. <script> element upisujemo unutar <head> ili <body> elementa.

N: Tako je – unutar <head> ili <body> elementa upisujemo <script> element i njegov atribut type sa vrijednosti *text/javascript*.

N: Kasnije ćemo kroz rad ponoviti još gradiva. Krenut ćemo sad sa rješavanjem zadatka. Prvo ću vam pokazati rješenje, a zatim ćemo ga zajedno rješavati. Neka vas ne začudi što ću osobno raditi u Notepad++ alatu – vi radite u Dreamweaver-u u kojem i inače radite na ovom predmetu. Također, testirat ćemo u više pretraživača – Google Chrome, Mozilla Firefox, Internet Explorer i Opera.

N: Koje pretraživače imate instalirane na svojim računalima?

**U**: [učenici odgovaraju koje pretraživače imaju instalirane]

**N:** U redu. Vi testirajte onda u onim pretraživačima koje imate instalirane.

N: Molim vas da sad otvorite alat u kojem ćete raditi – Dreamweaver. Kratica za alat vam se nalazi ili na Desktop-u, ili na Taskbar-u ili pak odete na Start > All Programs, pa potražite na listi programa. Kad
otvorite alat, vjerojatno će vam se automatski otvoriti i novi dokument, a ako se ne otvori, otvorite ga samostalno tako što ćete odabrati File > New.

**N:** Tko nije otvorio novi dokument, molim neka digne ruku.

# Glavni dio

U glavnom dijelu sata ću proći sa učenicima zadatke za vježbu. Odradit će se tri zadatka – dva vezana uz CSS3 i jedan vezan uz kombinaciju CSS3-a i JS-a. Zadaci će se rješavati cijeli prvi sat i trideset minuta drugog sata, nakon čega će započeti ponavljanje te će učenici dobiti samostalni zadatak.

# 1. zadatak

**N:** Rješenje našeg prvog današnjeg zadatka izgleda ovako *[demonstracija]*. Vidite dvije slike, svemir i raketu, a kad pređemo pokazivačem preko svemira, raketa leti iz donjeg lijevog ruba u gornji desni rub. Kad maknemo pokazivač sa svemira, raketa se vraća u svoj početni položaj. Zadatak ćemo rješavati korak po korak. Svoj rad spremajte u onu datoteku u kojoj vam je i materijal jer su vam ondje slike koje ćete pozivati u ovom zadatku.

N: Kada otvorite novi dokument, vama su automatski već otvoreni osnovni elementi. Spremite dokument kao *raketa.html*.

N: Upišite u naslov stranice *Mali korak...*. Unutar <head> elementa ćete otvoriti i zatvoriti i <style> element s atributom type i pripadnom vrijednosti *text/css*. Naš kod izgleda ovako, a kada pokrenemo u pretraživaču, stranica izgleda ovako [*demonstracija*].

N: Ako je netko zapeo, molim neka digne ruku.

N: Na rješenju ste mogli primijetiti kako imamo dva objekta – svemir i raketu. Tako ćemo i pristupiti rješavanju – unutar <body> elementa ćemo otvoriti i zatvoriti <div> i dodijelit ćemo mu atribut id koji ima vrijednost *svemir*. Kako se raketa nalazi u svemiru, tako ćemo unutar tog <div> elementa koji nam predstavlja svemir, otvoriti i zatvoriti <div> kojem ćemo dodijeliti atribut class koji ima vrijednost *raketa*. Unutar <div> elementa koji nam predstavlja raketa. Unutar <div> elementa koji nam predstavlja raketu, ćemo otvoriti i zatvoriti još jedan <div> bez ikakvih atributa. Naš kod izgleda ovako, a kada pokrenemo u pretraživaču, stranica izgleda ovako [*demonstracija*].

N: Prije nego što krenemo na uređivanje i animiranje, pitanje – koja je razlika između atributa id i class?

U: Pomoću id i class atributa opisujemo sadržaj u HTML dokumentu. Pomoću njih možemo dohvatiti HTML elemente – id označava samo jedan element, odnosno, pomoću njega možemo dohvatiti samo jedan element jer je jedinstven, dok isti class može imati više elemenata u istom dokumentu i na taj način možemo istovremeno dohvatiti više elemenata kojima je vrijednost tog atributa ista.

N: Tako je – atribute id i class koristimo da bismo dohvatili elemente. id je jedinstven i može ga imati samo jedan element, odnosno, vrijednost id atributa je jedinstvena za svaki element, dok istu vrijednost atributa class može imati više elemenata istovremeno.

N: Nastavljamo. Naš daljnji rad će se bazirati uglavnom na CSS-u i radit ćemo unutar našeg <style> elementa koji se nalazi unutar <head> elementa. Kao što je već rečeno, postoji svemir i raketa unutar tog svemira. Unutar <style> elementa ćemo stoga upisati ono što ćemo i urediti. Nakon što smo upisali što moramo urediti, naš kod izgleda ovako [demonstracija].

N: Uređivat ćemo redom. Za početak ćemo u div#svemir dodati svojstva za širinu, visinu i dodat ćemo pozadinsku sliku. Sve slike vam se nalaze u materijalima u datoteci 1\_zadatak. Kod izgleda ovako, a kada pokrenemo u pretraživaču, stranica izgleda ovako [demonstracija].

N: Dalje ćemo poslati raketu u taj svemir tako što ćemo urediti div.raketa div element. Definirat ćemo visinu, širinu i pozadinsku sliku. Kod izgleda ovako, a kada pokrenemo u pretraživaču, stranica izgleda ovako [demonstracija].

N: Da bismo pozicionirali raketu dolje lijevo, trebamo urediti div.raketa element. Dodajemo svojstva za pozicioniranje i redom dolje i lijevo. Kod izgleda ovako, a kada pokrenemo u pretraživaču, stranica izgleda ovako [demonstracija].

N: Primijetite kako se naša raketa ne nalazi u svemiru, nego izvan njega. Kako možemo postaviti raketu da ipak bude u svemiru?

U: Pomoću svojstva position kojem ćemo vrijednost postaviti na relative.

N: Tako je – tako ćemo i napraviti. Jednostavno ćemo u div#svemir dodati svojstvo position sa vrijednosti *relative*.

N: Koja je razlika između absolute i relative vrijednosti svojstva position?

**U:** Kada svojstvo position postavimo na *relative*, elementi su pozicionirani relativno u odnosu na margine dokumenta, a kada ga postavimo na *absolute*, može doći do preklapanja elemenata jer su elementi postavljeni na točno određene pozicije na stranici.

**N:** Tako je – svojstvo position općenito određuje način pozicioniranja bloka. Vrijednost mu može biti *relative* i *absolute*. Kada ga postavimo na *relative*, elementi su pozicionirani relativno u odnosu na margine dokumenta – u našem primjeru, svemir mora biti u gornjem lijevom rubu dokumenta. A kada ga postavimo na *absolute*, može doći do preklapanja elemenata jer su elementi postavljeni na točno određene pozicije na stranici – u našem primjeru smo htjeli da nam raketa bude u svemiru i postavili smo to tako da se elementi na kraju preklapaju.

N: Sada imamo raketu u svemiru, ali ništa se ne događa. Mi želimo animirati svoju raketu da leti i da se rotira kada se pokazivačem pređe preko svemira, a da se vrati u početni položaj kad se pokazivač makne sa svemira. Da bismo to napravili, unutar svog <style> elementa ćemo dodati još div#svemir:hover div.raketaidiv#svemir:hover div.raketa div.

N: Naša se raketa mora okrenuti i pomaknuti. Gdje ćemo koristiti vrijednost *rotate*, a gdje vrijednost *translate* atributa transform?

U: Raketu mičemo tako da pomičemo div#svemir:hover div.raketa, a okrećemo je tako što okrećemo div#svemir:hover div.raketa div.*rotate* ćemo koristiti u okretanju rakete, dakle u div#svemir:hover div.raketa div, dok ćemo translate koristiti u pomicanju rakete i to u div#svemir:hover div.raketa.

N: Tako je. Dakle, da bismo okrenuli raketu, u div#svemir:hover div.raketa div ćemo dodati svojstvo transform i vrijednost *rotate*, a da bismo je pomaknuli ćemo u div#svemir:hover div.raketa dodati svojstvo transform i vrijednost *translate*. Nakon toga, naš kod izgleda ovako, a kada pokrenemo u pretraživaču, stranica izgleda ovako [demonstracija].

N: Zašto sam upisala više puta sličnu liniju koda koristeći prefikse -webkit-, -moz-, -o-, -ms-?

U: Zato što ne podržavaju svi pretraživači jednako CSS3.

**N:** Tako je. U materijalima možete vidjeti dokument *zanimljivi\_linkovi.txt*. Između ostalog, možete pronaći link na kojem ćete vidjeti na koji način koje svojstvo je podržano od kojeg pretraživača.

N: Naša raketa sad skače, a trebala bi putovati u gornji desni kut. To ćemo postići tako što ćemo unutar div.raketa i div.raketa div dodati svojstvo transition.

N: Što radi svojstvo transition?

U: Svojstvo transition služi kako bismo postepeno elementu promijenili stil.

N: Koja je razlika između vrijednosti *ease-in* i *ease-in-out*?

**U:** *ease-in* određuje svojstvu transition spori početak, dok *ease-in-out* određuje svojstvu transition spori početak i kraj.

N: Tako je – ako želimo promijeniti jedan stil u drugi istom elementu, koristimo svojstvo transition. ease-in i ease-in-out određuju spori početak tog svojstva ako se radi o prvom, odnosno, sporom početku i kraju svojstva ako se radi o drugom.

N: Naš kod sada izgleda ovako, a kada pokrenemo u pretraživaču, stranica izgleda ovako [demonstracija].

N: Ovo je ujedno i naše rješenje zadatka. Ako netko nije gotov, neka digne ruku. Molim vas da pokrenete u pretraživaču svoje rješenje [*provjeravanje jesu li svi učenici gotovi*].

## 2. zadatak

N: Sada ćemo se prebaciti na JavaScript. Rješenje našeg sljedećeg zadatka izgleda ovako [demonstracija] – razbacani pravokutnici. Nakon što se klikne na neki pravokutnik, oni se pomiješaju. Za početak ćete otvoriti novi dokument, dodati <script> element s atributom type i njegovom vrijednosti text/javascript. Naslov stranice je Mijesanje pravokutnika..., a kad spremate dokument, spremite ga kao mijesanje.html. Za sad nam kod izgleda ovako, a kada pokrenemo u pretraživaču, stranica izgleda ovako [demonstracija].

N: Prvo ćemo riješiti stilove. Unutar <style> elementa ćemo dodati div.pravokutnik, a svojstva će redom biti background, color, width, height, line-height, text-align i position. Odmah ćemo dodati i animaciju koja će kasnije omogućiti da nam pravokutnici ne skaču nego plivaju po ekranu. Dodajemo svojstvo transition. CSS3 kod izgleda ovako [demonstracija].

N: Sad ćemo se pozabaviti JavaScriptom. Zadat ćemo dvije globalne varijable i tri funkcije. Varijable su spremi i ukupno. Funkcije koje ćemo upisati će biti goForIt(), shakeEmUp() i getNewPoint(). Naš JavaScript kod nakon što smo upisali globalne varijable i funkcije izgleda ovako [demonstracija].

N: Redom ćemo upisati kod u naše funkcije. Prva funkcija koju ćemo napraviti je getNewPoint() kojom određujemo poziciju svakog <div> elementa, odnosno, pravokutnika na ekranu i to tako što ćemo vratiti izračunatu vrijednost top (po visini) i left (po širini). Naš kod izgleda ovako [demonstracija].

N: rx:Math.round(Math.random() \* 800) označava gdje će se <div> nalaziti po širini (po xosi).ry:Math.round(Math.random() \* 600) označava gdje će se <div> nalaziti po visini (po y-osi).

- 40 -

N: Slijedeća funkcija koju ćemo stvoriti je goForIt() koja stvara i raspoređuje <div> elemente po ekranu. Za početak je potrebno definirati varijable root, divy i pt. Varijabla root nam govori gdje će se cijeli kod izvršavati, divy je varijabla koja sprema <div> elemente, a pt je varijabla koja će spremati funkciju getNewPoint(). Nakon definiranja varijabli, stvorit ćemo for petlju unutar koje ćemo kreirati <div> elemente te kojima ćemo dodijeliti atribut id, upisat ćemo im tekst pomoću innerHTML-a te ćemo im dodati atribut class. Nakon toga ćemo pozvati funkciju getNewPoint() i spremiti je u varijablu pt te ćemo svakom <div> elementu dodijeliti atribut style sa pripadajućim vrijednostima za left i top poziciju. Nakon toga ćemo prikazati sve <div> elemente i spremiti ih u niz.

N: Koliko će sveukupno biti <div> elemenata i gdje to definiramo?

U: Sveukupno će biti 100 <div> elemenata, a broj definiramo u globalnoj varijabli ukupno.

N: Tako je – u globalnoj varijabli ukupno definiramo maksimalni broj <div> elemenata. Naš kod će izgledati ovako [demonstracija].

N: <div> elemente kreiramo sa divy = document.createElement('div');. Definiramo atribut id sa divy.setAttribute("id", "sudDiv" + i);. Broj koji se ispisuje u pravokutniku se određuje sa divy.innerHTML = i + 1;. Vrijednost atributa class definiramo sa divy.className = "pravokutnik";. Funkciju koja pozicionira <div> elemente nasumično po ekranu spremamo u varijablu pt, a zatim svakom <div> elementu dodajemo atribut <style> sa left i top vrijednostima. Na kraju prikazujemo sve <div> elemente koje smo stvorili i spremamo ih u niz.

N: Za kraj, riješit ćemo zadnju funkciju – shakeEmUp(). Ova funkcija mijenja poziciju svakog <div> elementa nakon klika na bilo koji od njih. Prvo ćemo dodati jednu for petlju unutar koje ćemo pozvati ponovno funkciju getNewPoint() i spremiti je u varijablu pt. Nakon toga ćemo svim spremljenim <div> elementima promijeniti top i left vrijednost i to u pikselima. Kod izgleda ovako, a kada pokrenemo u pretraživaču, stranica izgleda ovako [demonstracija].

N: Zašto ne radi?

U: Zato što nismo pozvali funkcije.

N: Tako je – da bi proradilo, moramo pozvati funkciju koja će razbacati naše <div> elemente, a to je goForIt() i funkciju koja će na klik pomiješati <div> elemente i promijeniti im poziciju, a to je pak shakeEmUp().

#### N: Kako ćemo to napraviti?

U: <body> elementu ćemo dodati onLoad i onClick događaje - na onLoad će se pozvati funkcija
goForIt(), a na onClick funkcija shakeEmUp().

N: Dobro, a koja je razlika između onLoad i onClick?

U: onLoad – događaj se aktivira kada je dokument učitan. onClick – događaj se aktivira kada korisnik klikne na element.

N: Tako je – onLoad koristimo kada želimo da se događaj aktivira čim je dokument učitan (u našem slučaju je to funkcija goForIt()), a onClick kada događaj želimo aktivirati nakon što korisnik klikne na element (u našem slučaju to je funkcija shakeEmUp()).

N: Kada dodate onLoad i onClick, naš kod izgleda ovako, a kada pokrenemo u pretraživaču, stranica izgleda ovako [demonstracija].

**N:** To je ujedno i kraj ovog zadatka. Kada kliknemo na bilo koji <div> element, svi pravokutnici promijene poziciju. Ako netko nije gotov, molim neka digne ruku. Molim vas da pokrenete u pretraživaču svoje rješenje [*provjeravanje jesu li svi učenici gotovi*].

# 3. zadatak

N: Rješenje našeg sljedećeg zadatka izgleda ovako [demonstracija]. Ovdje su izreke koje se otvaraju nakon što kliknemo na naslov. Za početak otvorite novi dokument i spremite dokument kao harmonika.html. Naslov koji upisujete unutar <title> elementa je Izreke. Naš kod izgleda ovako [demonstracija].

N: Unutar <body> elementa ćemo otvoriti i zatvoriti <div>. Tom <div> elementu ćemo dodati atribut class čija je vrijednost *harmonika*. Unutar tog <div> elementa ćemo upisati ostatak našeg koda – taj <div> možemo shvatiti kao određenu kantu u kojoj se sve nalazi. Prvo ćemo dodati naslov <h2> i vrijednost *lzreke*. Nakon toga ćemo ispod naslova dodati šest <div> elementata sa idiclass atributima – id neka budu redom vrijednosti *jedan, dva, tri, cetiri, pet, sest*, a atribut class neka u svim <div> elementima ima vrijednost *sadrzaj*.

N: Zbog bolje preglednosti ćemo prije svakog <div> elementa dodati komentar sa brojem <div> elementa, kako bismo znali s kojim <div> elementom radimo.

N: Kako se umeću komentari u HTML, a kako u CSS?

**U**: U HTML se upisuje upisuje otvorena izlomljena zagrada, uskličnik, dva minusa kao početak komentara (<!--), a kao kraj komentara dva minusa i zatvorena izlomljena zagrada (-->). U CSS-u se komentar označava sa kosa crta i zvjezdica kao početak komentara (/\*) te kao zvjezdica i kosa crta kao kraj komentara (\*/).

N: Tako je – ako želimo u HTML-u staviti komentar koristimo otvorenu izlomljenu zagradu, uskličnik i dva minusa za početak komentara, te dva minusa i zatvorenu izlomljenu zagradu za kraj komentara. U CSS-u koristimo kosu crtu i zvjezdicu za početak komentara te zvjezdicu i kosu crtu za kraj komentara. Nakon što smo to napravili, naš kod izgleda ovako, a kada pokrenemo u pretraživaču, stranica izgleda ovako [demonstracija].

N: Unutar svakog tog <div> elementa dodajte naslov <h3>, unutar kojeg je element <a> s atributom href čija je vrijednost *#broj* gdje je broj redom *jedan, dva, tri, cetiri, pet, sest* ovisno o id atributu <div> elementa, te ispod naslova dodajte jedan prazan <div>. Naš kod sada izgleda ovako [demonstracija].

**N:** Ako netko nije došao do ovog dijela, molim neka digne ruku. Ako ste svi došli do ovdje, onda otvorite dokument *izreke.txt* koji vam se nalazi u materijalima u datoteci 2\_zadatak i upišite tekst u svoj dokument – ono što je pod jednostrukim navodnicima upišite u naslov, a tekst koji se nalazi ispod upišite u prazan <div> element koji ste stvorili. Imate šest izreka i šest <div> elemenata. Naš kod izgleda ovako, a kada pokrenemo u pretraživaču, stranica izgleda ovako [demonstracija].

**N:** Taj umetnuti tekst ćemo urediti koristeći se i <br /> elementima.

N: Što označava element, a što <br /> element?

U: element definira odlomak, a <br /> element stvara prekid linije.

N: Tako je – sa elementom stvaramo odlomke, a tekst, odnosno linije teksta prekidamo <br /> elementom.

N: Sad ćemo urediti tekst. Prva izreka je unutar jednog elementa. Druga izreka također ima samo jedan element, a nakon tri točkice se nalazi <br /> element. Treća izreka ima jedan element.
Četvrta izreka ima dva elementa – u drugom odlomku nakon tri točkice se nalazi <br /> element.
Peta izreka je unutar jednog elementa. Šesta izreka je unutar jednog elementa.

N: Naš kod izgleda ovako, a kada pokrenemo u pretraživaču, stranica izgleda ovako [demonstracija].

N: Sad krećemo na uređivanje CSS-a. Unutar <style> elementa ćemo dodati redom elemente koje ćemo i uređivati.

N: Što znači dio h3 + div?

U: h3 + div znači da se odabiru svi <div> elementi koji su postavljeni odmah nakon <h3> elementa.

N: Što znači dio :target h3 + div?

U: :target h3 + div znači da će biti odabran trenutni aktivni <div> element koji se nalazi ispod <h3> elementa.

N: Tako je – kad naiđete na element1 + element2 u CSS-u, to znači da se stilovi odnose na sve element2 koji se nalaze odmah nakon element1. :target h3 + div se odnosi na <div> element koji se nalazi ispod naslova koji smo odabrali – dakle, klikom na naslov, otvara nam se točno taj <div> element i niti jedan drugi, a koji ima svoju visinu i ostala svojstva. U *zanimljivi\_linkovi.txt* možete pronaći i link za CSS selektore.

N: div.harmonika elementu ćemo promijeniti pozadinsku boju, dodat ćemo mu obrub, padding svojstvo, širinu i *auto* margin. div.harmonika div.sadrzaj elementu ćemo dodati donji obrub, padding svojstvo i pozadinsku boju. Kod izgleda ovako, a kada pokrenemo u pretraživaču, stranica izgleda ovako [demonstracija].

N: Koja je razlika između margin i padding svojstva?

U: margin svojstvo u CSS-u stvara vertikalne i horizontalne prostore između elemenata. Ako elementi nemaju definiran margin oko sebe, bit će zalijepljeni jedni na druge. Neki elementi imaju standardne vrijednosti svojstva margin, pa se često, iako se nisu definirale, one mogu primijetiti oko elemenata.

U: padding svojstvo u CSS-u je horizontalni i vertikalni prostor koji se nalazi oko sadržaja odabranog elementa. padding je unutar elementa, a ne izvan kao što je margin.

N: Tako je – evo i primjera. Na ovoj slici možete primijetiti da je margin prostor izvan elementa, a padding prostor unutar elementa, odnosno, od sadržaja do ruba koji je često nevidljiv [demonstracija].

N: Što se događa kad upišemo margin: auto;?

U: Kad upišemo margin i postavimo mu vrijednost na *auto*, element se postavlja na sredinu njegovog nadređenog elementa.

N: Tako je – ako postavimo na *auto*, kao što smo mi to napravili u našem primjeru, taj element će se postaviti u sredinu njegovog roditelja, odnosno, nadređenog elementa.

N: Zadnja dva elementa nam omogućuju efekt padajućeg izbornika. U div.harmonika h3 + div element ćemo dodati svojstvo height i overflow svojstvo ćemo postaviti na *hidden*. U div.harmonika :target h3 + div element ćemo svojstvo height, a overflow svojstvo ćemo postaviti na *auto*. Kod izgleda ovako, a kada pokrenemo u pretraživaču, stranica izgleda ovako [demonstracija].

N: Čemu služi svojstvo overflow?

U: Svojstvo overflow određuje što će se dogoditi ako sadržaj prijeđe preko prostora elementa.

N: Tako je – nama to svojstvo treba kako bismo dobili efekt da je tekst koji nije odabran ostane sakriven i zato je postavljen na *hidden*, odnosno, na *auto* kad kliknemo na element za koji želimo da se prikaže tekst.

N: Naš padajući izbornik sada ne pada nego skače. Da bismo postigli animaciju, unutar div.harmonika h3 + div ćemo dodati svojstvo transition.

N: Ovo smo svojstvo već koristili ali smo umjesto vrijednosti height koristili all – koja je razlika?

U: Kada smo koristili *all*, mijenjala su se sva svojstva, a kada sad koristimo *height*, mijenja se samo visina elementa.

N: Tako je – *all* je značilo da ćemo promijeniti sva svojstva elementa, ali možemo mijenjati i pojedinačno svojstva. U ovom slučaju to je visina.

N: Naš kod izgleda ovako, a kada pokrenemo u pretraživaču, stranica izgleda ovako [demonstracija].

**N:** I to je ujedno i naše rješenje zadatka. Ako netko nije gotov ili ima neki problem, molim neka digne ruku. Molim vas da pokrenete u pretraživaču svoje rješenje [*provjeravanje jesu li svi učenici gotovi*].

# Završni dio

U završnom dijelu ću sa učenicima ponoviti što smo radili te ću im zadati samostalni zadatak. Za samostalni zadatak je predviđeno deset minuta.

N: Danas smo prošli osnovne elemente - <html>, <head>, <body> - u Dreamweaver-u se ti elementi automatski stvore kada se otvori novi dokument. Kada se otvori novi dokument u Dreamweaver-u, automatski se stvori još jedan element – koji?

U: Stvori se <title> element.

N: Tako je - <title>. Čemu služi taj element?

U: U njega upisujemo naziv stranice.

N: Tako je – element <title> koristimo kako bismo upisali naziv stranice.

N: Važno je prilikom pisanja HTML koda svaki otvoreni tag i zatvoriti. Na što prilikom toga moramo paziti?

U: Naziv elementa koji smo otvorili treba biti jednak elementu koji zatvaramo.

N: Tako je – onaj naziv koji smo upisali u otvoreni tag moramo upisati u zatvoreni tag.

N: Danas smo u zadacima koristili CSS3 i JS. Na koji način smo umetali CSS i JS u naš HTML kod?

U: U <head> elementu smo koristili <script> element ako je u pitanju JS, odnosno <style> element ako je u pitanju CSS3, unutar kojeg smo zatim upisivali JS odnosno CSS kod.

N: Tako je – koristili smo takozvani interni način umetanja CSS-a i JS-a te smo u <head> element ubacivali prema potrebi <style> i <script>.

N: Kada smo uređivali našu raketu, svemir i izreke – kako smo dohvaćali te elemente? Koje smo HTML atribute koristili?

**U:** Koristili smo id i class atribute.

N: Tako je – idiclass. Koja je razlika između njih?

U: id označava samo jedan element, odnosno, pomoću njega možemo dohvatiti samo jedan element jer je jedinstven, dok isti class može imati više elemenata u istom dokumentu i na taj način možemo istovremeno dohvatiti više elemenata kojima je vrijednost tog atributa ista.

N: Tako je – id je jedinstven i može ga imati samo jedan element, dok istu vrijednost atributa class može imati više elemenata istovremeno.

N: Što radi CSS svojstvo transition?

U: Svojstvo transition služi kako bismo postepeno elementu promijenili stil.

N: Tako je – pomoću svojstva transition postepeno mijenjamo stil elementu.

N: Što radi CSS svojstvo transform?

U: Svojstvo transform služi kako bismo promijenili element, odnosno, da bismo ga rotirali, mijenjali mu veličinu, pomicali i slično.

N: Tako je – pomoću transform svojstva mi mijenjamo naš element. Mi smo danas koristili vrijednosti rotate i translate – rotate da bismo rotirali element, a translate da bismo pomaknuli element.

N: Zašto koristimo u CSS3-u prefikse -webkit-, -moz-, -o-, -ms-?

U: Zato što ne podržavaju svi pretraživači jednako CSS3.

N: Tako je – razne verzije pretraživača razno podržavaju CSS3.

N: Koja je razlika između onLoad i onClick?

U: onLoad – događaj se aktivira kada je dokument učitan. onClick – događaj se aktivira kada korisnik klikne na element.

N: Tako je – onLoad koristimo kada želimo da se događaj aktivira čim je dokument, a onClick kada događaj želimo aktivirati nakon što korisnik klikne na element.

# 4. zadatak

N: Odlično. Sad kad smo ponovili, možete riješiti svoj samostalni zadatak. Na papir ili na računalu u programu za uređivanje teksta kao što je npr. Word ćete zapisati linije koda koje ste ispravili. U datoteci *4\_zadatak* u materijalima koje ste preuzeli, imate dokument *lampa\_greske.html*. Kad taj dokument otvorite u pretraživaču, vidite samo prazan ekran. To je zato što taj kod u sebi ima greške. Sveukupno je dvadesetipet grešaka i na vama je da ih pronađete i ispravite kako biste vidjeli kako izgleda krajnje rješenje. Da vam malo olakšam, greške su uglavnom izostavljena slova, zagrade i slično. Sav je kod tu, ništa ne nedostaje i ne morate ništa mijenjati – samo ispraviti tipfelere. Na kojoj god liniji koda nađete grešku, zapišite broj te linije da mogu dok prolazim vidjeti kako vam ide i jeste li nešto možda preskočili. Kada ste gotovi me pozovite da pogledam.

**N**: Molim vas da prije nego što izađete iz učionice ugasite laptope i poklopite ih i uredite svoje radno mjesto. Molim vas da ne ostavljate smeće na stolu i oko stola. Hvala.

N: Doviđenja, ugodan dan vam želim!

U: Doviđenja!

#### Kraj pripreme

## **1.4.** Primjer pisane pripreme 4. – javno predavanje

Sljedeća priprema pisana je za izvođenje nastavnog sata na **Prvoj gimnaziji Varaždin**. U realizaciji nastave su sudjelovali učenici drugog razreda **umjerenja opće gimnazije**. Obrađivana tema, odnosila se na programiranje u C++ programskom jeziku. Za primijetiti je koliko se ova priprema razlikuje od pripreme za izvođenje nastavnog sata na **Elektrostrojarskoj školi Varaždin**. Naime po ovom obrazovnom programu, učenici su dobili puno više teorijskih osnova u realizaciji rješenja zaddataka od na primjer učenika **Elektrostrojarske škole** gdje se programiranje u C++ programskom jeziku odnosilo na praktičnu primjenu i rješenje problema.(*usporedi s Primjer pisane pripreme 2, str. 18*)

#### Početak pripreme:

Škola: Prva gimnazija , Varaždin

Razred: 2f. razred

## Nadnevak izvođenja nastavnog sata: 12.svibnja 2014.

Nastavna jedinica: Polja (nizovi)

Nastavna cjelina: Jednodimenzionalna polja (nizovi)

Tip sata:1. sat: obrada novog gradiva

2.sat: vježbanje gradacijom zadataka

#### Cilj nastavnog sata:

Upoznati učenike s programskim mogućnostima rada s jednodimenzionalnim poljima

#### Materijalni zadaci:

- Kombinirati ostalu funkcionalnost C jezika u radu s jednodimenzionalnim poljima
- Primijeniti funkcije ispisa u jednodimenzionalnim poljima
- Primijeniti funkcije upisa u jednodimenzionalnim poljima
- Izraditi programske zadatke upotrebom jednodimenzionalnih polja.

#### Funkcionalne zadaci:

- Razvijati dobru orijentaciju unutar okruženja C jezika
- Osposobljavati učenike za razumijevanje praktične primjene jednodimenzionalnih polja
- Poticati kreativan pristup rješavanju konkretnog zadatka u C jeziku promjenom jednodimenzionalnog polja
- Poticati interes za samostalno otkrivanje ostalih mogućnosti primjene jednodimenzionalnih polja

## Odgojni zadaci:

- Razvijati smisao za red i organizaciju sadržaja
- Stjecati pozitivan stav prema radu
- Razvijati suradnju i međusobno pomaganje

## Nastavna pomagala:

- Računalo
- Projektor

## Nastavne metode:

- Metoda razgovora
- Metoda usmenog izlaganja
- Metoda demonstracije
- Metoda praktičnog rada

## Oblici rada:

- Frontalni rad
- Samostani rad

## Literatura:

- Tatjana Stranjak, Vesna Tomić, C-jezik, udžbenik za drugi i treći razred srednje škole, Zagreb,
   2005. Školska knjiga
- Nina Lipljin, Programiranje 1, Varaždin, 2004. Tiva Tiskara
- B. Motik, J.Stribar, Demistificirani C++, 2010., Element

Koraci nastavnog procesa i trajanje u minutama		Sadržaj koraka
Uvodni dio sata	8 minuta	Pozdrav i predstavljanje Upoznavanje s pravilima i načinom rada Postavljanje nekoliko početnih pitanja Upoznavanje učenika sa sadržajem današnjeg nastavnog sata
Glavni dio sata 1. sat	37 minuta	Upoznavanje s jednodimenzionalnim poljima Demonstriranje ispisa elemenata u jednodim. poljima Demonstriranje unošenja vrijednosti u jednodim. polja Ponavljanje putem rješavanja zadataka
Kratki uvodni dio 2. sata	2 minute	Kratki osvrt na prethodni sat i napomena o radu na drugom satu

Glavni dio sata 2. sat	37 minuta	<ul> <li>Izrada zadataka:</li> <li>zadatak 1 (deklaracija polja )</li> <li>zadatak 2( ispisivanje elemenata )</li> <li>zadatak 3(ispis upotrebom for petlje)</li> <li>zadatak 4( unos upotrebom for petlje)</li> <li>zadatak 5(korištenje if else uvjetovanja)</li> <li>zadatak 6(dodavanje polja)</li> <li>zadatak 7(ispis obrnutim redoslijedom)</li> <li>zadatak 8( unos n elemenata i izračunavanje artirm. sredine)</li> <li>zadatak 9(ispis sume elemenata na parnim indeksima)</li> </ul>
Završni dio sata	6 minuta	Ponavljanje Podsjećanje učenika da isključe računala i pospreme radna mjesta Pozdrav

#### **TIJEK IZVOĐENJA NASTAVE**

#### Uvod (8 minuta)

N: Dobar dan! Moje je ime X.X. i danas ću biti vaš nastavnik. Molio bih vas da tijekom rada komuniciramo na način da podignete ruke i tek kada budete prozvani postavite svoje pitanje ili dajte adekvatan odgovor. Prije postavljenog pitanja recite svoje ime radi kvalitetnije komunikacije. Ukoliko će vam se činiti u nekom trenutku da idem prebrzo, prekinite me uz podizanje ruke i ja ću usporiti. Na vrijeme ćete biti obaviješteni kada dođe vrijeme odmora. Za početak uključite računala i pokrenite okruženje za rad u C jeziku. Provjerite da li se okruženje za rad u C-u nalazi na radnoj površini (ako da onda ga pokrenite), ako se ne nalazi na radnoj površini, potom potražite pod start i programs.

N: Cilj današnjeg nastavnog sata je upoznati jednu novu kategoriju s kojom se niste susretali do sada u programiranju, a to su jednodimenzionalna polja ili nizovi. Upoznat ćemo niz mogućnosti koje nude jednodimenzionalna polja, a ujedno ćete isto tako primjeni svoje dosadašnje znanje iz programiranja na jednodimenzionalna polja.

#### ASOCIJACIJSKA PITANJA VEZANA ZA POLJA (NIZOVE) :

N: U matematici ste sigurno radili nizove. Što su nizovi (polja) u matematici?

U:Nizovi u matematici predstavljaju nabrajanja brojeva u konačnom ili beskonačnom nizu.

N:Kako preciznije u matematici definiramo nizove?

U: Svaku funkciju kojoj je domena skup prirodnih brojeva zovemo nizom.

N:Koje vrste nizova (polja) ste do sada upoznali u matematici?

U: Nizovi mogu biti konstantan niz, Fibonaccijev niz, aritmetički niz, geometrijski niz.

N:Odlično. Dobar odgovor.

N:Gdje u svakodnevnom životu susrećete nizove?

U: Abeceda predstavlja niz, izbornik na računalu je isto tako niz.

N:Takve nizove praktički susrećete svugdje gdje postoji određena organizacija.

N:Što mislite iz kojih razloga se nizovi upotrebljavaju u svakodnevnom životu?

U:Da bi stvorili nekakav red, redoslijed, organizaciju te da bi lakše pristupali pojedinim sadržajima.

N:Tako je. Veći naglasak je na lakšem pristupanju velikim količinama sadržaja.

#### Glavni dio sata: 1. Sat (37 minuta)

N:Pomoću polja ili nizova, moguće je pristupiti većem broju podataka odjednom uporabom nekog simboličnog imena. To simbolično ime u programiranju zovemo indeksom. Ukoliko biste prilikom svojeg redovitog učenja htjeli posjetiti neke internetske izvore ili izvore digitalne knjižnice, pristupit ćete pretraživačima gdje se na temelju ključne riječi ili indeksa, izlista lista autora i naslova njihovih djela, koji su dohvaćeni upravo pomoću upisane ključne riječi.

N: Na sličan način pristupat ćemo podacima koji se nalaze pohranjeni pod jednim imenom ili indeksom u našem C jeziku.

#### (Krećem s prezentacijom "Metod1\_zoran\_hercigonja\_jednodimenzionalno\_polje")

N: Kao što sam rekao polje je niz varijabli istog tipa imenovanih zajedničkim identifikatorom, koje se nalaze na uzastopnim memorijskim lokacijama. Svaka varijabla u polju se naziva element polja. Jednodimenzionalno polje je najjednostavniji oblik polja čiji se elementi dohvaćaju s pomoću samo jednog indeksa. Elementi polja nalaze se u linearnom slijedu.

N: Da bi bili u mogućnosti deklarirati jednodimenzionalno polje, moramo poštovati sintaksu deklaracije. Sintaksa je sljedeća: *tip\_elemenata\_polja*- ili tip\_podataka koji predstavlja jedan od osnovnih tipova podataka u C-u.

N: S kojim tipovima podataka ste se do sad susretali?

U:Dosad smo se susretali s tipovima podataka char, int, float.

N: Tako je. To su osnovni tipovi podataka koji se mogu primjenjivati na elementima polja. Osim tipova podataka u deklaraciji imamo i *ime*-ili identifikator i *duljinu polja-* ili broj\_elemenata polja je najveći mogući broj elemenata koji se mogu upisati u dano polje.

(Demonstriram nad slajdom broj 6 prezentacije "Metod1\_zoran\_hercigonja\_jednodimenzionalno\_polje" postojanje indeksa s dodatnim komentarima koji su uvaženi na prezentaciji.

N: Kod svakog jednodimenzionalnog polja, važno je spomenuti da deklaracijom jednodimenzionalnog niza (polja) u memoriji se za svaki element rezervira posebna memorijska lokacija. Rezervirane lokacije obično idu jedna iza druge slijedno.

N:Da li varijable polja pripadaju istom ili različitom tipu podataka.

U:Varijable u polju su sve istog tipa.

N:Tako je. Varijable polja su određene tipom podataka kojim je deklarirano polje.

N: Ako deklariramo niz od 7 elemenata program prevoditelj osigurava u memoriji neprekinut prostor za 7 realnih podataka.

(Demonstriram nad slikom iz prezentacije Metod1\_zoran\_hercigonja\_jednodimenzionalno\_polje, slajd 11)



N: Ako se niz deklarira kao globalna varijabla onda je početna vrijednost svih elemenata niza jednaka 0.

N:Što znači da je varijabla deklarirana globalno? Gdje se nalazi u programskom kodu; ispred ili iza main funkcije?

U: Ispred main funkcije.

N: Ako se deklarira kao lokalna varijabla onda elementi niza nisu inicijalizirani. Ako je potrebno, polje se može inicijalizirati prilikom deklaracije ili u programu.

N: Ako prilikom inicijalizacije skup vrijednosti ima manje elemenata nego što je deklarirano, vrijednost preostalih elemenata je 0.

(Na slajdu 14 prezentacije "Metod1\_zoran\_hercigonja\_jednodimenzionalno\_polje" prikazujem mali odsječak programskog koda za kojeg tražim od učenika da ga prepišu i isprobaju prema mojim uputama )**Trajanje:2 minute** 

```
#include<stdio.h>
    int main(void)
{
        int b[7]={10, 20, 30, 40, 50};
        printf("\n %d", b[0]);
}
```

N: Pod printf polje odredite da ispisuje b[5] ili b[6]. Trebalo bi vam ispisivati nulu.

(Paralelno dok učenici prepisuju kod, ja prođem razredom provjeravajući njihov rad te isto provjeravam da se učenici ne bave kakvim drugim aktivnostima koje nisu primjerene za nastavu (npr. dopisuju se preko facebooka ili se koriste tabletom, mobitelom ili kakvim sličnim napravama ili nekim drugim sadržajima tipa časopis). Ukoliko se učenik bavi aktivnostima koje nisu primjerene za nastavu, provodim odgovarajuću disciplinsku mjeru.)

N: Jeste li svi uspjeli isprobati ovaj kod? Ako netko nije molim podignite ruke.

(Ukoliko netko nije uspio interveniram na određeno mjesto, no ako je više učenika podiglo ruke, onda se vraćam na isti zadatak i ponovno prolazim kroz njega s dodatnim napomenama. Ako je sve u redu, pokažem rješenje programa i ponovim ključne dijelove uz dodatno demonstraciju putem slike sa slajda 15 prezentacije "Metod1\_zoran\_hercigonja\_jednodimenzionalno\_polje").

b[0]	b[1]	b[2]	b[3]	b[4]	b[5]	b[6]
10	20	30	40	50	0	0

N: broj elemenata unutra polja može biti jednak veličini polja ili manji od te veličine polja.

N: Da vidimo sljedeći slučaj. Što se događa kada upišemo više elemenata od dopuštene veličine polja?

N: Prepišite programski odsječak i testirajte.

#include <stdio.h></stdio.h>
int main(void)
{
int b[3]={10, 20, 30, 40, 50};

## Trajanje:2 minute

(Paralelno dok učenici prepisuju kod, ja prođem razredom provjeravajući njihov rad te isto provjeravam da se učenici ne bave kakvim drugim aktivnostima koje nisu primjerene za nastavu (npr. dopisuju se preko facebooka ili se koriste tabletom, mobitelom ili kakvim sličnim napravama ili nekim drugim sadržajima tipa časopis). Ukoliko se učenik bavi aktivnostima koje nisu primjerene za nastavu, provodim odgovarajuću disciplinsku mjeru. )

N: Jeste li svi uspjeli isprobati ovaj kod? Ako netko nije molim podignite ruke.

}

(Ukoliko netko nije uspio interveniram na određeno mjesto, no ako je više učenika podiglo ruke, onda se vraćam na isti zadatak i ponovno prolazim kroz njega s dodatnim napomenama. Ako je sve u redu, pokažem rješenje programa i ponovim ključne dijelove.

N: Već u samom programskom editoru prilikom kompajliranja vam javlja moguću pogrešku u radu.
N: Duljina niza može se i izostaviti. Program prevoditelj će tada sam odrediti duljinu niza prema broju elemenata navedenih unutar vitičastih zagrada.

(Demonstriram ponovno zadajući učenicima da na svojim računalim isprobaju kod sa slajda 19)

Trajanje:2 minute

#include <stdio.h></stdio.h>	
	int main(void)
{	
	int b[]={10, 20, 30, 40, 50};
	printf("\n %d", b[5]);
}	

(Paralelno dok učenici prepisuju kod, ja prođem razredom provjeravajući njihov rad te isto provjeravam da se učenici ne bave kakvim drugim aktivnostima koje nisu primjerene za nastavu (npr. dopisuju se preko facebooka ili se koriste tabletom, mobitelom ili kakvim sličnim napravama ili nekim drugim sadržajima tipa časopis). Ukoliko se učenik bavi aktivnostima koje nisu primjerene za nastavu, provodim odgovarajuću disciplinsku mjeru.)

N: Jeste li svi uspjeli isprobati ovaj kod? Ako netko nije molim podignite ruke.

(Ukoliko netko nije uspio interveniram na određeno mjesto, no ako je više učenika podiglo ruke, onda se vraćam na isti zadatak i ponovno prolazim kroz njega s dodatnim napomenama. Ako je sve u redu, pokažem rješenje programa i ponovim ključne dijelove.)

N:Primijetite da ste do sada ispis elemenata polja radili upisujući indeks pod kojim se nalazila vrijednost koja vas je zanimala. Kada ste stavili printf ("%d", b[2]), ispisala se vrijednost koja se nalazi na trećem mjestu u polju pod indeksom 2.

N: Kada bi željeli napraviti ispis svih elemenata polja, kombinirali bi naredbu za ispis s programskim petljama koje omogućuju pomicanje po nizu promjenom indeksa. Od tri programske petlje najpraktičnija je for petlja.

N: Što mislite zašto je for najpraktičnija?

U: Zato što ima brojač i mogućnosti manipulacije indeksa putem brojača.

N:Odličan odgovor. Upravo se iz tih razloga primjenjuje for petlja.

(Predstavljam i ujedno ponavljam dekalraciju for petlje)

- **for** (kv=0; kv<n; kv++)
- printf (,,\n format", ime\_polja[kv]);
- **kv** kontrolna varijabla
- **n**-veličina niza (najveći broj elemenata)
- format-format za ispis kojiovisi o tipu elemenata upisanih u niz

(Zadajem mali odsječak programksog koda da učenici isprobaju funkciju ispisa uz pomoću for petlje) **Trajanje:2 minute** 

```
#include<stdio.h>
int main (void)
{
    int i, b[]={1, 2, 3};
    for(i=0; i<3; i++)
    printf ("\n %d", b[i]);
}</pre>
```

(Paralelno dok učenici prepisuju kod, ja prođem razredom provjeravajući njihov rad te isto provjeravam da se učenici ne bave kakvim drugim aktivnostima koje nisu primjerene za nastavu (npr. dopisuju se preko facebooka ili se koriste tabletom, mobitelom ili kakvim sličnim napravama ili nekim drugim sadržajima tipa časopis). Ukoliko se učenik bavi aktivnostima koje nisu primjerene za nastavu, provodim odgovarajuću disciplinsku mjeru. )

N: Jeste li svi uspjeli isprobati ovaj kod? Ako netko nije molim podignite ruke.

(Ukoliko netko nije uspio interveniram na određeno mjesto, no ako je više učenika podiglo ruke, onda se vraćam na isti zadatak i ponovno prolazim kroz njega s dodatnim napomenama. Ako je sve u redu, pokažem rješenje programa i ponovim ključne dijelove.)

(Otvaram slajd 23 iz prezentacije "Metod1\_zoran\_hercigonja\_jednodimenzionalno\_polje" i zadajem zadatak)

N: Bez prepisivanja koda u okruženje C jezika i kompajliranja, možete li reći što će ispisati sljedeći program.

U:Program će ispisati vrijednosti: 4, 8, 12, 16, 0.

N:Dobro, a zašto ispisuje baš te vrijednosti?

U: Zbog postavljenih uvjeta u for petlji.

N: Što to znači?

U: Kontrolna vrijednost u petlji jednaka je 1 što znači da će prvi element imati indeks 1 to jest dat će broj 4.

N:Tako je. Kako bi protumačili ostala postavljena ograničenja u for petlji?

U: Vrijednost kontrole varijable uvećava se za 2 pa se svaki drugi element preskače. Prilikom deklaracije polja zauzeto je 10 mjesta, a prilikom inicijalizacije popunjeno je samo 8 . Deveti i deseti element automatski se popunjava nulom.

N: Tako je to se sve vrlo jasno izmanipuliralo for petljom. Sada uviđate njezin značaj u radu s jednodimenzionalnim poljima.

N: Kao i kod ispisa vrijednosti, za unošenje nove vrijednosti koriste se programske petlje. I isto tako najkorisnija je for petlja.



printf (,,\n format", &ime\_polja[kv]);

N: Primjetite deklaracija je ista kao kod ispisa, samo je dodan operator &

N:Što je i čemu služi & operator?

U: To je operator adrese i vraća adresu varijable koja se nalazi iza njega.

N:Tako je.

(Zadajem mali odsječak programksog koda da učenici isprobaju funkciju upisa uz pomoću for petlje)

Trajanje:2 minute

(Paralelno dok učenici prepisuju kod, ja prođem razredom provjeravajući njihov rad te isto provjeravam da se učenici ne bave kakvim drugim aktivnostima koje nisu primjerene za nastavu (npr. dopisuju se preko facebooka ili se koriste tabletom, mobitelom ili kakvim sličnim napravama ili nekim drugim sadržajima tipa časopis). Ukoliko se učenik bavi aktivnostima koje nisu primjerene za nastavu, provodim odgovarajuću disciplinsku mjeru.) N: Jeste li svi uspjeli isprobati ovaj kod? Ako netko nije molim podignite ruke.

(Ukoliko netko nije uspio interveniram na određeno mjesto, no ako je više učenika podiglo ruke, onda se vraćam na isti zadatak i ponovno prolazim kroz njega s dodatnim napomenama. Ako je sve u redu, pokažem rješenje programa i ponovim ključne dijelove.)

## Ponavljanje gradiva putem rješavanja programskih zadataka.

#### Zadatak 1(trajanje 3 minute)

- Deklarirajte polje od 10 elemenata
- a[10]={2, 4, 6, 8, 10, 12, 14, 16, 18, 20};

(Paralelno dok učenici prepisuju kod, ja prođem razredom provjeravajući njihov rad te isto provjeravam da se učenici ne bave kakvim drugim aktivnostima koje nisu primjerene za nastavu (npr. dopisuju se preko facebooka ili se koriste tabletom, mobitelom ili kakvim sličnim napravama ili nekim drugim sadržajima tipa časopis). Ukoliko se učenik bavi aktivnostima koje nisu primjerene za nastavu, provodim odgovarajuću disciplinsku mjeru.)

N: Jeste li svi uspjeli isprobati ovaj kod? Ako netko nije molim podignite ruke.

(Ukoliko netko nije uspio interveniram na određeno mjesto, no ako je više učenika podiglo ruke, onda se vraćam na isti zadatak i ponovno prolazim kroz njega s dodatnim napomenama. Ako je sve u redu, pokažem rješenje programa i ponovim ključne dijelove.)

## Zadatak 2 (trajanje 3 minute)

- Deklarirajte polje od 10 elemenata
- a[10]={2, 4, 6, 8, 10, 12, 14, 16, 18, 20};
- I ispišite 1, 4, 6 i 10 element

N: Prisjetimo se. Što nam u nizovima pokazuje poziciju elemenata?

U: Indeksi.

#### N:Točno.

- N: U zadatku piše ispišite prvi element polja. Kolja je početna vrijednost indeksa?
- U: Početna vrijednost indeksa je 0.

N: Dobro. Možete rješavati zadatak.

(Paralelno dok učenici prepisuju kod, ja prođem razredom provjeravajući njihov rad te isto provjeravam da se učenici ne bave kakvim drugim aktivnostima koje nisu primjerene za nastavu (npr. dopisuju se preko facebooka ili se koriste tabletom, mobitelom ili kakvim sličnim napravama ili nekim drugim sadržajima tipa časopis). Ukoliko se učenik bavi aktivnostima koje nisu primjerene za nastavu, provodim odgovarajuću disciplinsku mjeru.)

N: Jeste li svi uspjeli isprobati ovaj kod? Ako netko nije molim podignite ruke.

(Ukoliko netko nije uspio interveniram na određeno mjesto, no ako je više učenika podiglo ruke, onda se vraćam na isti zadatak i ponovno prolazim kroz njega s dodatnim napomenama. Ako je sve u redu, pokažem rješenje programa i ponovim ključne dijelove.)

## Zadatak 3 (trajanje 3 minute):

 Nadogradite zadatak 2 tako da deklarirate novo polje od 10 elemenata b[10]; i u njega napravite unos od 5 elemenata

N: Prisjetimo se što se dogodi ako prilikom inicijalizacije skup vrijednosti ima manje elemenata nego što je deklarirano?

U: vrijednost preostalih elemenata su 0?

N: Tako je možete rješavati zadatak.

(Paralelno dok učenici prepisuju kod, ja prođem razredom provjeravajući njihov rad te isto provjeravam da se učenici ne bave kakvim drugim aktivnostima koje nisu primjerene za nastavu (npr. dopisuju se preko facebooka ili se koriste tabletom, mobitelom ili kakvim sličnim napravama ili nekim drugim sadržajima tipa časopis). Ukoliko se učenik bavi aktivnostima koje nisu primjerene za nastavu, provodim odgovarajuću disciplinsku mjeru.)

N: Jeste li svi uspjeli isprobati ovaj kod? Ako netko nije molim podignite ruke.

(Ukoliko netko nije uspio interveniram na određeno mjesto, no ako je više učenika podiglo ruke, onda se vraćam na isti zadatak i ponovno prolazim kroz njega s dodatnim napomenama. Ako je sve u redu, pokažem rješenje programa i ponovim ključne dijelove.)

#### Zadatak 4(trajanje 4 minute)

Nadogradite zadatak 3 tako da upotrebom for petlje napravite ispis elemenata polja a[10]={2,
4, 6, 8, 10, 12, 14, 16, 18, 20};

(Paralelno dok učenici prepisuju kod, ja prođem razredom provjeravajući njihov rad te isto provjeravam da se učenici ne bave kakvim drugim aktivnostima koje nisu primjerene za nastavu (npr. dopisuju se preko facebooka ili se koriste tabletom, mobitelom ili kakvim sličnim napravama ili nekim drugim sadržajima tipa časopis). Ukoliko se učenik bavi aktivnostima koje nisu primjerene za nastavu, provodim odgovarajuću disciplinsku mjeru.)

N: Jeste li svi uspjeli isprobati ovaj kod? Ako netko nije molim podignite ruke.

(Ukoliko netko nije uspio interveniram na određeno mjesto, no ako je više učenika podiglo ruke, onda se vraćam na isti zadatak i ponovno prolazim kroz njega s dodatnim napomenama. Ako je sve u redu, pokažem rješenje programa i ponovim ključne dijelove. )

#### Školsko zvono i vrijeme odmora

N: Sada je vrijeme za odmor i zvonilo je. Možete na 5 minuta izaći i malo se osvježiti.

(Drugi dio sata je osmišljen kao provođenje vježbi odnosno rad s učenicima kroz postepeno rješavanje zadataka od onih najlakših do onih najtežih. Na početku sata oduzmem 2 minute za kratku uputu što ćemo raditi taj drugi sat. Cijeli drugi nastavni sat vodim po prezentaciji "Zadaci\_za\_vjezbu\_cetvrti\_sat")

#### Uvodni dio: (2 minute)

N:Na prethodnom satu smo samo prošli kroz mogućnosti jednodimenzionalna polja kroz deklaraciju polja, funkcije unosa, ispisa, te sortiranja. Sada ćemo na konkretnim primjerima deklarirati polja i iskušati sve funkcionalnosti spomenute na prethodnom satu. Molim vas da tijekom rada svaki zadatak koji ćemo raditi stavite u novu source datoteku kako bi mogli lakše dograditi idući zadatak korištenjem prethodnog koda.

#### Glavni dio sata: 2. Sat (37 minuta)

N: Možete pokrenuti okruženje za C jezik i krenut ćemo s prvim zadatkom.

(Otvaram prezentaciju "Zadaci za vježbu" i zadajem prvi zadatak.)

## 1.Zadatak (trajanje 2 minute)

- Deklarirajte polje od 10 elemenata
- a[10]={2, 4, 6, 8, 10, 12, 14, 16, 18, 20};

(Paralelno dok učenici prepisuju kod, ja prođem razredom provjeravajući njihov rad uz određene intervencije po potrebi te isto provjeravam da se učenici ne bave kakvim drugim aktivnostima koje nisu primjerene za nastavu (npr. dopisuju se preko facebooka ili se koriste tabletom, mobitelom ili kakvim sličnim napravama ili nekim drugim sadržajima tipa časopis). Ukoliko se učenik bavi aktivnostima koje nisu primjerene za nastavu, provodim odgovarajuću disciplinsku mjeru.)

N: Jeste li svi uspjeli isprobati ovaj kod? Ako netko nije molim podignite ruke.

(Ukoliko netko nije uspio interveniram na određeno mjesto, no ako je više učenika podiglo ruke, onda se vraćam na isti zadatak i ponovno prolazim kroz njega s dodatnim napomenama. Ako je sve u redu, pokažem i komentiram rješenje programa uz osvrt na bitne ključne dijelove za dobivanje rješenja..)

N:U zadatku je trebalo napraviti unos elemenata ne pojedinačnim unošenjem elemenata nego elementima polja koji su definirani prilikom deklaracije polja.

N:Upotrijebite programski kod iz zadatka 1.

# 2.Zadatak (trajanje 2 minute)

- Deklarirajte polje od 10 elemenata
- a[10]={2, 4, 6, 8, 10, 12, 14, 16, 18, 20};
- I ispišite 1, 4, 6 i 10 element

(Paralelno dok učenici prepisuju kod, ja prođem razredom provjeravajući njihov rad uz određene intervencije po potrebi te isto provjeravam da se učenici ne bave kakvim drugim aktivnostima koje nisu primjerene za nastavu (npr. dopisuju se preko facebooka ili se koriste tabletom, mobitelom ili kakvim sličnim napravama ili nekim drugim sadržajima tipa časopis). Ukoliko se učenik bavi aktivnostima koje nisu primjerene za nastavu, provodim odgovarajuću disciplinsku mjeru.)

N: Jeste li svi uspjeli isprobati ovaj kod? Ako netko nije molim podignite ruke.

(Ukoliko netko nije uspio interveniram na određeno mjesto, no ako je više učenika podiglo ruke, onda se vraćam na isti zadatak i ponovno prolazim kroz njega s dodatnim napomenama. Ako je sve u redu, pokažem i komentiram rješenje programa uz osvrt na bitne ključne dijelove za dobivanje rješenja..) N: U zadatku je trebalo napraviti 4 ispisa pazeći na početni element polja. Na prethodnom satu sam spomenuo da je početni element polja uvijek nulti element.

N: Kada bi htjeli ispisati sve elemente polja odjednom na zaslon, zasigurno ne bi ručno ispisivali element po element. Ako imate polje od 1000 elemenata onda bi to bilo vrlo zamorno. Najbolje je da za tu vrstu ispisa upotrijebimo for petlju kao u sljedećem zadatku.

#### Zadatak 3 (trajanje 4 minute)

• Upotrebom for petlje napravite ispis svih elemenata polja iz zadatka 2

(Paralelno dok učenici prepisuju kod, ja prođem razredom provjeravajući njihov rad uz određene intervencije po potrebi te isto provjeravam da se učenici ne bave kakvim drugim aktivnostima koje nisu primjerene za nastavu (npr. dopisuju se preko facebooka ili se koriste tabletom, mobitelom ili kakvim sličnim napravama ili nekim drugim sadržajima tipa časopis). Ukoliko se učenik bavi aktivnostima koje nisu primjerene za nastavu, provodim odgovarajuću disciplinsku mjeru.)

N: Jeste li svi uspjeli isprobati ovaj kod? Ako netko nije molim podignite ruke.

(Ukoliko netko nije uspio interveniram na određeno mjesto, no ako je više učenika podiglo ruke, onda se vraćam na isti zadatak i ponovno prolazim kroz njega s dodatnim napomenama. Ako je sve u redu, pokažem i komentiram rješenje programa uz osvrt na bitne ključne dijelove za dobivanje rješenja..)

N: For petlju smo koristili za ispis elemenata polja u prethodnom zadatku. Sada ćemo istu petlju koristiti i za unos elemenata u polje. Molim da prilikom pisanja programa obratite pažnju na ključan identifikator koji nam omogućuje unos i pridruživanje.

## 4. Zadatak (trajanje 4 minute)

 Nadogradite zadatak 3 na način da unos elemenata polja a[] napravite putem for petlje

(Paralelno dok učenici prepisuju kod, ja prođem razredom provjeravajući njihov rad uz određene intervencije po potrebi te isto provjeravam da se učenici ne bave kakvim drugim aktivnostima koje nisu primjerene za nastavu (npr. dopisuju se preko facebooka ili se koriste tabletom, mobitelom ili kakvim sličnim napravama ili nekim drugim sadržajima tipa časopis). Ukoliko se učenik bavi aktivnostima koje nisu primjerene za nastavu, provodim odgovarajuću disciplinsku mjeru.) N: Jeste li svi uspjeli isprobati ovaj kod? Ako netko nije molim podignite ruke.

(Ukoliko netko nije uspio interveniram na određeno mjesto, no ako je više učenika podiglo ruke, onda se vraćam na isti zadatak i ponovno prolazim kroz njega s dodatnim napomenama. Ako je sve u redu, pokažem i komentiram rješenje programa uz osvrt na bitne ključne dijelove za dobivanje rješenja..)

N: Dakle o kojem je operatoru bila riječ prilikom unosa vrijednosti niza?

U: & operatoru ili operatoru adrese.

N: Tako je. Operator & vraća adresu varijable koja se nalazi iza njega.

N: U prethodnim zadacima uvjetovanje smo koristili putem for petlje. No postoji i drugi programski konstrukt koji omogućava uvjetovanje.

N: O kojem se konstruktu radi?

U: Radi se o if else uvjetovanju.

N: Tako je.

N: Za sljedeći zadatak trebat će nam uvjetovanje if else budući da ćemo elemente polja unositi pojedinačno.

## Zadatak 5. (trajanje 4 minute)

- 1. Ima izbornik kojim definirate broj elemenata u polju
- 2. printf ("upisite broj elemenata u polju:");
- 3. Uz pomoću if else ispituje uvjet da li je broj elemenata manji ili jednak 10 te da li polje nije prazno
- 4. if(n<=10 && n>0)
- 5. Ispisuje poruku o grešci ukoliko uvjet nije zadovoljen
- 6. Obavezno koristiti for petlje u realizaciji rješenja

N: Iskoristit ćemo prethodni 4 zadatak koji već ima definiran unos i ispis putem for petlje.

(Cijeli ovaj zadatak odradim zajedno s učenicima na način da pišem kod, učenici prepisuju kod i ja u međuvremenu objašnjavam napisane linije koda.)

(Nakon napisanog postavim pitanje: "Jeste li svi uspjeli? Ako ima netko tko nije uspio podignite molim ruke. Ukoliko se radi o jednom učeniku interveniram na to mjesto i dok pomažem tom učeniku, ostale upućujem na testiranje programskog primjera. Ako se radi o više učenika, ponavljam zadatak naglašavajući ključne elemente. U slučaju da su svi učenici uspjeli prepisati kod testiram programsko rješenje s komentarima oko tog rješenja.) N: Sljedeći zadatak (zadatak 6) podrazumijeva da deklarirate još jedno polje i primijenite sve elemente zadatka 5 na novo polje. Oprezno kopirajte dijelove koda i nemoj te zaboraviti staviti novu oznaku polja jer inače program neće raditi.

## Zadatak 6. (trajanje 3 minute)

- 1. Nadogradite zadatak 5 na način da:
- 2. Dodate još jedno polje b[10] te da nad njim primjenite upute iz zadatka 5

(Paralelno dok učenici prepisuju kod, ja prođem razredom provjeravajući njihov rad uz određene intervencije po potrebi te isto provjeravam da se učenici ne bave kakvim drugim aktivnostima koje nisu primjerene za nastavu (npr. dopisuju se preko facebooka ili se koriste tabletom, mobitelom ili kakvim sličnim napravama ili nekim drugim sadržajima tipa časopis). Ukoliko se učenik bavi aktivnostima koje nisu primjerene za nastavu, provodim odgovarajuću disciplinsku mjeru.)

N: Jeste li svi uspjeli isprobati ovaj kod? Ako netko nije molim podignite ruke.

(Ukoliko netko nije uspio interveniram na određeno mjesto, no ako je više učenika podiglo ruke, onda se vraćam na isti zadatak i ponovno prolazim kroz njega s dodatnim napomenama. Ako je sve u redu, pokažem i komentiram rješenje programa uz osvrt na bitne ključne dijelove za dobivanje rješenja..)

## Zadatak 7. (trajanje 5 minuta)

- Učitajte niz od N cijelih (realnih) brojeva te ispišite niz obrnutim redoslijedom učitavanja
- NAPOMENA: U realizaciji rješenja primjeniti while i do while petlju

(Cijeli ovaj zadatak odradim zajedno s učenicima na način da pišem kod, učenici prepisuju kod i ja u međuvremenu objašnjavam napisane linije koda.)

(Nakon napisanog postavim pitanje: "Jeste li svi uspjeli? Ako ima netko tko nije uspio podignite molim ruke. Ukoliko se radi o jednom učeniku interveniram na to mjesto i dok pomažem tom učeniku, ostale upućujem na testiranje programskog primjera. Ako se radi o više učenika, ponavljam zadatak naglašavajući ključne elemente. U slučaju da su svi učenici uspjeli prepisati kod testiram programsko rješenje s komentarima oko tog rješenja.)

## Zadatak 8. (trajanje 5 minuta)

N: Koristeći dijelove prethodnog koda izradite program za izračunavanje aritmetičke sredine.

N: Koji je matematički algoritam za izračunavanje aritmetičke sredine.

U: Aritmetička sredina se izračunava da se zbroje svi elementi niza i onda podijele s ukupnim brojem elemenata u nizu.

N:Dakle u programu trebate prvo trebate primijeniti operaciju zbrajanja, a onda operaciju dijeljenja. Isto tako trebate sumu ili zbroj inicijalizirati na vrijednost 0.

N: Pazite kod deklaracije da vam suma bude deklarirana decimalnim tipom podataka, jer aritmetička sredina ne mora nužno biti cijeli broj, već može biti decimalni broj.

- 1. Učitajte niz od N cijelih (realnih) brojeva
- 2. ispišite aritmetičku sredinu učitanog niza

(Paralelno dok učenici prepisuju kod, ja prođem razredom provjeravajući njihov rad uz određene intervencije po potrebi te isto provjeravam da se učenici ne bave kakvim drugim aktivnostima koje nisu primjerene za nastavu (npr. dopisuju se preko facebooka ili se koriste tabletom, mobitelom ili kakvim sličnim napravama ili nekim drugim sadržajima tipa časopis). Ukoliko se učenik bavi aktivnostima koje nisu primjerene za nastavu, provodim odgovarajuću disciplinsku mjeru.)

N: Jeste li svi uspjeli isprobati ovaj kod? Ako netko nije molim podignite ruke.

(Ukoliko netko nije uspio interveniram na određeno mjesto, no ako je više učenika podiglo ruke, onda se vraćam na isti zadatak i ponovno prolazim kroz njega s dodatnim napomenama. Ako je sve u redu, pokažem i komentiram rješenje programa uz osvrt na bitne ključne dijelove za dobivanje rješenja..)

## Zadatak 9. (trajnje 8 minuta)

N: Iskoristite dio koda iz zadatka 8 posebice onaj gdje se izvršava operacija zbrajanja. Dakle tu operaciju zbrajanja trebate prilagoditi da zbraja parne indekse.

N: Koji programski konstrukt ćemo uzeti za ispitivanjeparnosti indeksa?

U: If uvjetovanje.

(Otvaram prezentaciju "Metod1\_zoran\_hercigonja\_jednodimenzionalno\_polje\_treci\_sat" i dolazim do slajda 11 te objašnjavam parne indekse)



N:Tako je . Pomoću if uvjetovanja ćemo postaviti uvjet da se zbraja svaki indeks u brojaču koji podijeljen s 2 nema ostatka. To znači da će ostatak biti jednak nuli.

N: Ovdje ćemo koristiti upravo modularno dijeljenje zbog toga da bismo mogli usporediti da li je ostatak jednak nuli ili je različit od nje.

- 1. N: Iz toga slijedi: if(brojac%2==0){
- printf("indeks [ %d ] = %d\n",brojac,brojevi[brojac]);
- 3. }

(Otvaram slajd 20 prezentacije "Zadaci\_za\_vjezbu\_drugi\_cetvrti\_sat"i dajem im pomoć u kodu.)

- 1. Učitajte niz od N cijelih (realnih) brojeva
- 2. ispišite sumu elemenata s parnim indeksima

(Paralelno dok učenici prepisuju kod, ja prođem razredom provjeravajući njihov rad uz određene intervencije po potrebi te isto provjeravam da se učenici ne bave kakvim drugim aktivnostima koje nisu primjerene za nastavu (npr. dopisuju se preko facebooka ili se koriste tabletom, mobitelom ili kakvim sličnim napravama ili nekim drugim sadržajima tipa časopis). Ukoliko se učenik bavi aktivnostima koje nisu primjerene za nastavu, provodim odgovarajuću disciplinsku mjeru.)

N: Jeste li svi uspjeli isprobati ovaj kod? Ako netko nije molim podignite ruke.

(Ukoliko netko nije uspio interveniram na određeno mjesto, no ako je više učenika podiglo ruke, onda se vraćam na isti zadatak i ponovno prolazim kroz njega s dodatnim napomenama. Ako je sve u redu, pokažem i komentiram rješenje programa uz osvrt na bitne ključne dijelove za dobivanje rješenja..) (Zadaci 10, 11, 12 i 13, prezentacije "Zadaci\_za\_vjezbu\_drugi\_cetvrti\_sat" predstavljaju pričuvne zadatke ukoliko će učenic biti vrlo uspješni u rješavanju prethodnih zadataka[prema preporuci nastavnika Bojana Banića]).

## Zadnji dio sata (trajanje 6 minuta)

U zadnjih šest minuta ponovit ću s učenicima što smo zajedno prošli na današnjim predavanjima na oba nastavna sata).

N: Danas smo prošli kroz veći dio mogućnosti rada s jednodimenzionalnim poljima i praktičnim putem vidjeli smo čemu polja služe u informatici odnosno programiranju.

PITANJA ZA PONAVLJANJE:

N: Čemu služe polja u programiranju?

U: Polja nam služe da bismo lakše pristupali velikim količinama sadržaja.

N: Pomoću čega se pretražuju varijable u jednodimenzionalnom polju?

U: Varijable unutra polja pretražujemo pomoću indeksa.

N:Kakvu vrijednost indeksa ima prvi, a kakvu posljednji element polja?

U: Element polja s prvim indeksom ima vrijednost 0, a posljednji N-1.

- N: Kakvog su tipa podataka varijable u polju?
- U:Varijable u polju su sve istog tipa.
- N:Čime je određen tip podataka u polju?
- U: Tip podataka je određen tipom kojim je deklarirano polje.
- N: Nabrojite najjednostavnije oblike sortiranja koje smo prošli na satu?
- U:To su sortiranje zamjenom i sortiranje metodom mjehurića.

## (U posljednjoj minuti)

N: Upravo će zvoniti hvala na suradnji. Isključite računala i pospremite radna mjesta.

## N: Doviđenja.

U: Doviđenja.

## Školsko zvono.

## Kraj pripreme:
### **1.5.** Primjer pisane pripreme 5. – javno predavanje

Iduća priprema za izvođenje nastavnog sata, pisana je za nastavni sat održan na **Prvoj gimnaziji Varaždin.** Sadržaj se odnosio na obradu slike u Photoshop aplikaciji. Nasava je održana četvrtom razredu. Pristup sadržaju ove pripreme, usmjeren je na dob učenika. U prethodnoj pripremi, sadržaj je bio vrlo deteljiziran i prilagođen kognitivnim mogućnostima drugog razreda. Ovdje su učenici više samostalniji i u pripremi se više provodi samostalnog rada negoli vođene vježbe kao u prethodnoj pripremi.

Početak pripreme:

Škola: Prva gimnazija Varaždin

Razred: 4.

Nastavni predmet: Informatika

Nastavna cjelina: Web i multimedija

Nastavna jedinica: Photoshop

Tip sata: obrada novog gradiva

Blok sat: da

Datum: 10. travnja 2014.

# PLAN VOĐENJA (ORGANIZACIJE) NASTAVNOG PROCESA

### Cilj obrade nastavne jedinice

Upoznati učenike s alatom za obradu rasterske računalne grafike.

### Zadatci koje treba ostvariti da bi se postigao cilj

Materijalni zadatci:

- Opisati i usporediti tipove računalne grafike
- Uočiti razliku između formata pohrane slikovnih podataka
- Razlikovati i grupirati alate s obzirom na funkciju
- Pravilno procijeniti prikladnost primjene pojedinog alata

### Funkcionalni zadatci:

- Razvijati sposobnost analitičkog pristupa problemu
- Uvježbati korištenje osnovnih alata
- Razvijati stvaralačko ili kreativno mišljenje

### Odgojni zadatci:

- Poticati kreativnost u rješavanju zadataka
- Poticati međusobnu suradnju
- Razvijati smisao za estetiku

### Nastavne metode

- Metoda razgovora
- Metoda usmenog izlaganja
- Metoda demonstracije
- Metoda praktičnih aktivnosti

### **Oblici rada**

- Frontalni rad
- Individualni rad

### Nastavna sredstva i pomagala

- Računalo
- Projektor
- PowerPoint prezentacija
- Adobe Photoshop

### Izvori znanja

- Obrada teksta i slike: materijali s predavanja, auditornih i laboratorijskih vježbi
- Multimedijski sustavi: materijali s predavanja i vježbi
- Rudolph Musngi, A Quick History of Adobe Photoshop & Cool Facts Behind the Living Legend, 14. listopad 2013., <u>http://www.1stwebdesigner.com/inspiration/history-of-adobe-photoshop/</u> (dostupno 4. travnja 2014.)
- <u>http://en.wikipedia.org/wiki/Adobe\_Photoshop</u> (dostupno 4. travnja 2014.)

## ORGANIZACIJA NASTAVNOG RADA – ARTIKULACIJA METODIČKE JEDINICE (makroplan)

REDNI BROJ	FAZE RADA I SADRŽAJ	METODIČKO OBLIKOVANJE	VRIJEME U MINUTAMA
1.	<ul> <li>UVODNI DIO <ul> <li>Pozdravljanje i upoznavanje s učenicima (kratko predstavljanje da ću ja danas održati predavanje)</li> <li>Popis prisustvovanja (prozivanje redara da prijavi one koji danas nisu prisutni)</li> <li>Dogovor o načinu rada (javljanje dizanjem ruku)</li> <li>Predstavljanje sadržaja nastavne jedinice (popis aktivnosti koje će se odraditi)</li> </ul> </li> </ul>	<ul> <li>Usmeno izlaganje</li> <li>Razgovor</li> <li>Frontalni rad</li> </ul>	5'
2. 2.1.	<ul> <li>SREDIŠNJI DIO</li> <li>Obrada novog sadržaja 1 <ul> <li>Definiranje osnovnih pojmova (grafika, računalna grafika, rasterski i vektorski tipovi računalne grafike)</li> </ul> </li> <li>Kratka povijest Photoshop-a (prvotni autor(i), vlasnik, tržišni utjecaj, istaknute važnije verzije, aktualna verzija i promjena poslovnog modela, format)</li> </ul>	<ul> <li>Usmeno izlaganje</li> <li>Razgovor</li> <li>Demonstracija</li> <li>Praktična aktivnost</li> <li>Frontalni rad</li> </ul>	40'

	- Upoznavanje s alatom (izgled	
	sučelja, paleta alata)	
	- Prikladnost odabira formata	
2.2.		
	Vježba 1	
	- Otvaranje slike, izrezivanje i	
	spremanje u različite formate	
	(učenici isprobavaju različite	
	formate, njihove prednosti i	
	nedostatke, uspoređuju veličinu	
	izlazne datoteke)	
2.2		
2.3.	Obrada novog sadržaja 2	
	- Raspon tonova i boja (osnovni	
	koraci u podesavanju)	
2.4.		
	Vježba 2	
	- Spašavanje uništene fotografije	
	na kojoj prevladavaju sivi tonovi	
	i loša ravnoteža boja (korištenje	
	alata levels, curves, colour	
	balance, hue/saturation)	
	- Upotreba slojeva	
2.5.		
	Obrada novog sadržaja 3.1.	
	- uvod u retuširanje fotografije,	
	selekcije i Quick Mask način	
2.6.	rada	

	Vježba 3.1. - Uvod u vježbu i korištenje Quick Mask načina rada za promjenu boje očiju - Najava zvona		
3.	MALI ODMOR		
4.	SREDIŠNJI DIO	- Razgovor	35'
4.1.	Obrada novog sadržaja 3.2.	- Demonstracija	
	- Retuširanje fotografije (prikaz	- Praktična aktivnost	
	raznih alata)	- Frontalni rad	
		- Individualni rad	
4.2.	Vježba 3.2.		
	- Uklanjanje bora, piercing-a,		
	podočnjaka, pjegica, rasjekline		
	smanjivanje crvenila nosnica,		
	smanjivanje čeljusti		
4.3.	Vježba 4.		
	- korištenje mapa pomaka		
	(Displacement maps)		
4.4.	Vježba 5. (opcionalno)		
	- kompozicija (stapanje tri		
	fotografije u jednu)		

5.	ZAVRŠNI DIO	- Razgovor	10'
	<ul> <li>Ponavljanje (tipovi računalne grafike, igra prepoznavanja formata fotografija, funkcije alata u Fotoshop-u)</li> <li>Najavljivanje zvona</li> </ul>	- Frontalni rad	
	<ul> <li>- Napolnena da ucemer pospreme radna mjesta</li> <li>- Pozdravljanje</li> </ul>		

### **TIJEK IZVOĐENJA NASTAVE – NASTAVNI RAD (mikroplan)**

### 1. Uvodni dio (5 minuta)

U ovom dijelu sata se predstavljam učenicima, radim popis prisustvovanja, dogovaram način rada i učenicima predstavljam popis aktivnosti koji ćemo odraditi u nastavnoj jedinici.

N: Dobar dan!

U: Dobar dan!

N: Ja sam X.X.; student Fakulteta Organizacije i Informatike i danas ću vam održati blok sat nastave iz informatike. Danas ću biti vaš nastavnik.

Prije prezentacije današnjih aktivnosti radim popis prisustvovanja.

N: Molim redara da prijavi tko nije prisutan.

U: (Prijavljuje one koji nisu prisutni.)

Najavljujem današnju temu.

Na prezentaciji pokazujem aktivnosti koje ćemo odraditi kroz blok sat.

Učenike upoznajem s pravilima rada.

N: U slučaju da imate pitanje, želite odgovoriti na moje pitanje ili znate odgovor na pitanje koje je postavio drugi učenik, molim vas podignite ruku. Također vas molim da računala ne koristite u druge svrhe, osim za potrebe gradiva koje ćemo danas naučiti.

### 2. Središnji dio 1 (40 minuta)

U ovom dijelu se izmjenjuje obrada novog sadržaja kroz razgovor i usmeno izlaganje; rješavanje zadataka samostalno bez nastavnikove pomoći i kroz demonstracije.

### 2.1. Obrada novog sadržaja 1 (10 minuta)

Učenike postupno upoznajem s osnovnim pojmovima grafika, računalna grafika, rasterskim i vektorskim tipovima grafike kroz razgovor i ponavljanje gradiva (od piksela, bita, odnosa piksela i bita do rastera/bitmape; od grafičkih primitiva točka, pravac, krivulja do vektorske grafike)

N: Što je to grafika?

*U:* (*daju razne odgovore*) grafiti, plakati, fotografije, crteži pračovjeka u špiljama, zemljopisne karte, brojevi, simboli, geometrijski oblici ...

N: (ovisno o broju odgovora potičem učenike na davanje većeg broja odgovora neverbalnom i verbalnom komunikacijom)

N: (radim sintezu odgovora) Grafika je vizualna prezentacija na nekoj podlozi.

N: Sada kada smo definirali što je grafika općenito, logično se postavlja pitanje što je to računalna grafika. Što vi podrazumijevate pod pojmom računalna grafika?

*U: (daju razne odgovore) grafička korisnička sučelja, interaktivno crtanje pomoću računala, dizajniranje/projektiranje pomoću računala, simulacija, animacija, ...* 

N: (radim sintezu odgovora) Računalna grafika uključuje skoro sve osim zvuka, a stvoreno je pomoću računala. Višeznačnost pojma odnosi se na prezentaciju i manipulaciju slikovnih podataka pomoću računala, slike proizvedene pomoću računala, razne tehnologije koje se koriste pri izradi i obradi slika.

N: (navodim da je računalnu grafiku moguće podijeliti na dva tipa; postavljam pitanje: Koja?)

### U: (daju, ne daju ili odgovor nije točan)

N/alt1: (u slučaju da nema odgovora ili odgovor nije točan, postupno učenike dovodim do odgovora) Ako prenesete fotografiju s fotoaparata na računalo, tu fotografiju zumirate, zatim ponovno zumirate pri čemu zumiranje prelazi 100% i dalje nastavite zumirati... Što ćete vidjeti?

U/alt1: (daju različite odgovore) kvadratiće, piksele, ...

N/alt2: (po potrebi radim usporedbu između bitova i piksela; od grafičkih primitiva dolazim do vektorske grafike; uspoređujem rastersku grafiku sa slikanjem, a vektorsku grafiku s crtanjem)

N/alt3: (u slučaju točnog odgovora, ponavljam točan odgovor istodobno pohvaljujući učenika) Odlično! Točno! Bravo! (neka od pohvala) Rasterska i vektorska računalna grafika. N: Upravo je rasterska grafika, tj. fotografija ono što ćemo danas obrađivati koristeći alat Photoshop.

Zatim putem usmenog izlaganja, popraćeno prezentacijom predstavljam autore Photoshop-a, povijesni razvoj, ističem važnije verzije, navodim trenutačnu verziju i promjenu poslovnog modela.

Usmenim izlaganjem i PowerPoint prezentacijom objašnjavam razliku između formata slikovnih datoteka, njihove prednosti i mane.

Učenicima kažem da pokrenu aplikaciju Photoshop, ukratko predstavljam grupu alata i čemu služe. Učenike uvodim u Vježbu 1 koju će samostalno riješiti, nakon čega ću provjeriti rješenja.

### 2.2. Vježba 1 (5 minuta)

Ovom vježbom učenici rade refleksiju prezentiranog sadržaja o prikladnosti formata pohrane slikovnih podataka, a istodobno se upoznaju sa sučeljem Photoshop-a.

Vježba u kojoj se od učenika traži otvoriti slikovnu datoteku v1\_formati.bmp, nakon čega ju trebaju pohraniti u različitim formatima (jpg, gif, png). Nakon pohranjivanja učenici bi trebali primijetiti različitu veličinu datoteka, različit raspon boja.

Provjeravam jesu li svi učenici riješili zadatak i prelazim na obradu novog sadržaja.

### 2.3. Obrada novog sadržaja 2 (7 minuta)

Usmeno izlaganje popraćeno PowerPoint prezentacijom. Osnovna obrada fotografija kroz raspon tonova i boja. Učenicima ću kroz primjer tri fotografije različitog sadržaja s pripadajućim histogramima objasniti kako čitati histograme i koje osnovne korake je potrebno napraviti u provjeri i podešavanju raspona i boja. Navedeno će učenici isprobati na primjeru u vježbi 2.

### 2.4. Vježba 2 (8 minuta)

Ovu vježbu učenici izvode zajedno s nastavnikom. Vježba je postavljena u problemskom obliku. Fotografija v2\_tonovi.psd je uništena prevladavajućim sivim tonovima i lošom ravnotežom boja. Učenici trebaju prepoznati u čemu je problem i kao ga riješiti. Nastavnik kroz pitanja na koja učenici odgovaraju potiče učenike na razmišljanje (refleksija sadržaja usmenog izlaganja) i tako dolazi do odgovora i zajedno s učenicima rješava zadatak.



Slika 2.4.1. Izgled fotografije prije obrade

Prije rješavanja učenicima sugeriram da ne rade na početnom sloju, nego da naprave njegovu kopiju na kojoj će raditi podešavanje tonova i boja kako bi sačuvali originalnu fotografiju i kasnije mogli lakše uspoređivati original i obrađenu fotografiju bez potrebe za uključivanjem/isključivanjem svake pojedine akcije.

Postupak rješavanja:

- Provjeriti histogram (Levels iz palete, tipkovničkom kraticom ctrl + L ili iz izbornika Image →Adjustments → Levels)
- 2. Podesiti raspon tonova i boja na slici (Levels i Curves)
- 3. Podesiti ravnotežu boja (Hue/Saturation)



Slika 2.4.2. Izgled fotografije nakon obrade

### 2.5. Obrada novog sadržaja 3.1. (4 minute)

Usmenim izlaganjem popraćenim prezentacijom radim kratki uvod u retuširanje fotografija.

Objašnjavam da je danas upotreba Photoshop-a posebno popularna kod obrade portretne fotografije, ali da fotografi u želji da modeli izgledaju što ljepše često znaju pretjerati s retuširanjem tako da se kao krajnji rezultat često dobije plastični izgled lica koji nema teksturu kože.

Dajem prikaz trake s alatima i radim podjelu alata (grupaciju) s obzirom na njihovu funkciju.

Tako je alate moguće podijeliti u četiri grupe:

- 1. Odabir, rezanje, mjerenje
- 2. Retuširanje
- 3. Crtanje
- 4. Navigacija

Govorim da ćemo se u prvom dijelu vježbe upoznati s izradom selekcija. Navodim da postoje četiri tipa alata za selekciju, a to su Marquee (alati za odabir na temelju definiranih (geometrijskih) oblika), Lasso (alati za odabir nepravilnih), Magic Wand i Quick Selection (alati

za izbor na temelju vrijednosti boja). Osim ovih alata navodim da je selekcije moguće raditi i uz pomoć Quick Mask koristeći alate koji nisu namijenjeni za selekciju.

Napominjem da ako je potrebno selekciju je moguće oduzimati (uz tipku shift), dodavati (uz tipku alt), i deselektirati (ctrl + D, Select  $\rightarrow$  Deselect).

### 2.6. Vježba 3.1. (6 minuta)

Ovu vježbu nastavnik radi zajedno s učenicima. Cilj prvog dijela vježbe je upoznati učenike s razlogom popularnosti Quick Mask načina rada s obzirom da je moguće odabrati dijelove slike bez alata za selekciju koristeći olovke, kistove i sl.

U ovom dijelu vježbe će učenici na fotografiji lice.psd promijeniti boju očiju iz smeđe u neku od boja po želji.

Prije izrade vježbe učenicima napominjem da se prilikom retuširanja slika preporučuje uvijek kreirati nove slojeve u paleti LAYERS u kojima se vrše korekcije slike, kako bi se pikseli slike sačuvali na jednom sloju, a promjene se tada mogu raditi neovisno na drugim slojevima.



Slika 2.6.1 Boja očiju prije obrade

Postupak rješavanja:

- 1. Otvoriti fotografiju v3\_lice.psd
- 2. Prebaciti se u Quick Mask način rada (tipka Q ili odabrati iz trake s alatima)
- 3. Foreground boju postaviti na crnu (iz trake s alatima ili iz palete color)

- 4. Odabrati Brush alat (tipka B ili odabir iz trake s alatima)
- 5. Podesiti veličinu kista da bude jednaka veličini oka
- 6. Lijevom tipkom miša škljocnuti na oba oka
- 7. Vratiti se u normalni način rada (tipka Q ili odabir iz trake s alatima)
- 8. Invertirati selekciju (Shift + Ctrl + I ili Select  $\rightarrow$  Inverse)
- Po potrebi poboljšati selekciju oduzimanjem (Alt) ili dodavanjem (Shift) nekim od alata za selekciju
- 10. Opcionalno omekšati selekciju za 2 piksela. (Shift + F6 ili Select → Modify → Feather)
- 11. Snimiti selekciju pod nazivom 'oci'. (Select  $\rightarrow$  Save selection)
- 12. Pomoću Hue/Saturation uključivši opciju Colorize obojiti oči u boju po želji (iz palete
   Adjustments ili Ctrl + U ili iz izbornika Image → Adjustments → Hue/Saturation)
- 13. Spremiti dosadašnji rad



Slika 2.6.2 Boja očiju nakon obrade

Provjeravam jesu li svi uspješno napravili i najavljujem pauzu.

### 3. Mali odmor

Pauza između blok sata je predviđena u trajanju 5 minuta.

- 3. sat (09:10 9:55)
- 4. sat (10:00 10:45)

### 4. Središnji dio 2 (35 minuta)

### 4.1. Obrada novog sadržaja 3.2. (5 minuta)

S učenicima ponavljam što smo do sada napravili u trećoj vježbi i što ćemo još napraviti. Upoznavanje s alatima za retuširanje Spot Healing Brush, Healing Brush, Dodge, Burn, Sponge i filterom Liquify.

Kroz kratak opis (jedna rečenica) navodim funkciju pojedinog alata.

Primjer. Spot Healing Brush je alat namijenjen ispravljanju manjih sitnijih nepravilnosti, a radi na principu da uzima vrijednosti okolnih piksela i te vrijednosti primjenjuje na piksele koje želimo ispraviti. Ukoliko želimo veću kontrolu onda je bolje koristiti Healing Brush s obzirom da sami definiramo uzorak koji želimo stopiti s odredišnim pikselima. Ukoliko nešto želimo posvijetliti moguće je koristiti alat Dodge, a ukoliko želimo suprotan efekt (potamnjivanje) koristimo alat Burn. Alat Sponge ima funkciju pojačavanja/smanjivanja intenziteta boje ovisno o odabranom parametru Mode (Saturate/Desaturate)

### 4.2. Vježba 3.2. (25 minuta)

Ova vježba je nastavak vježbe 3.1. Učenici rješavaju vježbu na način da nastavnik pokaže jedan primjer za svaki od koraka rješavanja, a učenici nastavljaju rješavati ostatak samostalno.

Postupak rješavanja:

- 1. Ukloniti pjegice i sitne točkaste nepravilnosti uz pomoć alata Spot Healing Brush
- 2. Ukloniti piercing iz nosa. (alat Spot Healing Brush)
- 3. Pomoću alata Healing Brush ispraviti pukotinu na donjoj usni.
- 4. Ukloniti podočnjake (alat Healing Brush)
- 5. Posvijetliti područje podočnjaka (Dodge alat koji posvjetljava)

- 6. Zatamniti pozadinu iza osobe. (Burn alat koji zatamnjuje)
- 7. Pojačati boju usana. (alat Sponge Saturate)
- 8. Smanjiti crvenilo nosnica (alat Sponge Desaturate)
- 9. Stanjiti čeljust, povećati usne i oči (filter Liquify)



Slika 4.2. Prikaz fotografije prije i nakon obrade

Provjeravam jesu li učenici riješili zadatak i radim uvod u vježbu 4.

### 4.3. Vježba 4. (5 - 7 minuta)

Korištenje mape pomaka (Displacement maps). Zadana dva sloja: naborana tkanina i zastava,. Potrebno je izraditi vijugavu zastavu koristeći naboranu tkaninu kao mapu pomaka za zastavu.

Postupak rješavanja:

- 1. Otvoriti datoteku v4\_mape\_pomaka.psd
- 2. Duplicirati sloj tkanine (iz palete LAYERS ili iz izbornika Layer → Duplicate Layer)

- Zamutiti sliku tkanine (želimo zadržati samo nabore tkanine, a ne teksturu) (Filter → Blur → odabrati npr. Gaussian Blur i postaviti vrijednost zamućenja)
- 4. Duplicirati tkaninu u novi dokument (Image  $\rightarrow$  Duplicate)
- Zamutiti sliku tkanine (želimo zadržati samo nabore tkanine, a ne teksturu) (Filter → Blur → odabrati npr. Gaussian Blur i postaviti vrijednost zamućenja)
- 6. Spremiti takvo zamućenu tkaninu u .psd dokument (npr. pomak.psd)
- U izvornom dokumentu na sloju zastave primijeniti filtar Displace (Filter → Distort → Displace i odabrati željenu mapu pomaka - pomak.psd)
- Na dupliciranom sloju tkanine promijeniti Blend Mode iz Normal u Hard Light. (iz palete LAYERS ili iz izbornika Layer → Layer Style → Blend Mode: Hard Light)



Slika 4.3. Izgled zastave prije i nakon primjene mape pomaka

### 4.3. Vježba 5. (opcionalno)

Za učenike koji su ranije završili s izradom vježbe tri, a da se ne bi dosađivali predviđen je dodatni zadatak. Pomlađivanje osobe (v5\_pomlađivanje.psd - bojanje sijede kose koristeći neki od alata (preporuka Quick Mask i Levels zato što je najbrže i najjednostavnije) i izglađivanje bora (preporuka upotreba nekih od filtera Noise npr. Median i alat Healing Brush)), ili izrada kompozicije (klavir.jpg, note.jpg, violinist.jpg - tri fotografije koje je potrebno stopiti u jednu)

### 5. Završni dio (8 - 10 minuta)

Ponavljanje

N: Koji tipovi računalne grafike postoje?

U: Rasterska i vektorska.

N: Koja je razlika između rasterske/bitmap i vektorske računalne grafike?

U: Rasterska računalna grafika sačinjena od mreže piksela, a vektorska od grafičkih primitiva točka, pravac, krivulja.

N: Photoshop je alat za obradu kojeg tipa računalne grafike?

U: Rasterske

Igra otkrivanja formata slikovnih datoteka.

Učenicima pokazujem tri slike istog sadržaja s različitim karakteristikama prozirnosti, dubine boja, veličine datoteke, a na učenicima je otkriti o kojim formatima je riječ. (refleksija zadatka 1)

Učenicima pokazujem fotografiju koju treba retuširati navodeći što bi trebalo popraviti, a od učenika se traži odgovor kojim alatom bi se to moglo napraviti. (refleksija zadatka 3) Najavljujem zvono. Zahvaljujem učenicima na suradnji i napominjem da pospreme svoje radno mjesto prije izlaska iz učionice.

Kraj pripreme

# 2. Individualna predavanja-srednje škole

Dosad smo imali prilike vidjeti čak pet priprema za javna predavanja kojima su nazočili metodičar, nastavnik-mentor i kolege studenti. Na individualnim prerdavanjima nastavi nazoči samo nastavnik-mentor. Ovdje se nastavnik-početnik više oslanja na sebe i svoje iskustvo stečeno na javnim predavanjima. U ovom poglavlju će biti prikazane dvije iznimno kvalitetne pripreme za izvođenje nastavnog sata.

### 2.1. Primjer pisane pripreme 1. –individualno predavanje

Priprema je pisana za izvođenje nastavnog saata na **Prvoj gimnaziji Varaždin** za učenike **drugog razreda općeg usmjerenja**. Područje informatike obuhvaćalo je programiranje odnosno algoritam sortiranja u programskom jeziku C++.

# Priprema za izvođenje nastavnog sata

Nastavnik:		X.X	Nadnevak:	11.04.2014.
Škola:		Prva gimnazija Varaždin	Razred:	1. razred
Nastavni predmet:		Informatika		
Nastavna cjelina:		Obrada teksta (Mi	icrosoft Word)	
Nastavna jedinica:		Unos i rad s tablicama u Microsoft Wordu		
Cilj nastavne jedinice:		Upoznavanje učenika s unosom i oblikovanjem tablica u Microsoft Wordu		
Tip nastavnog sata:		Obrada novog gradiva		
Zadaci: Naučiti kreirati tablice Naučiti kako se označavaju retci, stupci, tal Naučiti kako se mijenja pozadinska boja će Naučiti promijeniti veličinu tablice, stupaca Naučiti poravnati tekst u ćelijama Naučiti umetati i brisati retke i stupce Naučiti spajati i dijeliti ćelije u tablici Naučiti kako se dijeli tablica		oci, tablice i ćelije oja ćelija supaca i redaka e		
	Funkcionalni:	Razvijati sposobnost samostalnog rada u Microsoft Wordu i sličnim programima		
	Odgojni:	Poticati samostalnost Poticati osjećaj za preglednost, red i urednost Poticati pozitivne stavove prema radu		
Nastavne metode:		Metoda razgovora Metoda usmenog izlaganja Metoda demonstracije Metoda praktičnih aktivnosti		
Socijalni oblici rada:		Frontalna nastava Individualni rad		
Nastavna sredstva i pomagala:		Računalo Projektor		
Korelacija s drugim predmetima:		Učenici će novostečena znanja moći koristiti u drugim predmetima kroz izradu referata, eseja, seminara i ostalih samostalnih zadataka		
Izvori znanja:		Gvozdanović, Ikica, Kos, Milijaš, Milijaš, Srnec, Sekulić- Štivčević, Zvonarek (2009) INFORMATIKA ZA GIMNAZIJE – udžbenik s DVD-om za 1. i 2. razred te za izbornu nastavu informatike općih, jezičnih i klasičnih gimnazija. PRO-MIL d.o.o., Varaždin		

# Organizacija nastavnog rada – artikulacija metodičke jedinice (makroplan)

Koraci nastavnog procesa i trajanje		Sadržaj koraka	Socijalni oblici rada i
u minutama			nastavne metode
Uvodni dio sata	5	Pozdrav i predstavljanje	Frontalna nastava
		Upoznavanje s pravilima i	Metoda razgovora
		načinom rada	
			Metoda usmenog
		Ponavljanje predznanja iz	izlaganja
		Microsoft Worda	
		Upoznavanje učenika sa	
		sadržajem današnjeg nastavnog	
		sata	
Središnii dio sata	35	Pokazivnie unosa tablica te	Frontalna nastava
, <b>,</b>		oblikovanje istih uz rješavanje	
		zadatka paralelno:	Individualni rad
		- kreiranie tablica	Metoda razgovora
		- označavanje redaka stupaca	Matada usasanaa
		tablica i ćelija te promjena	Metoda usmenog
		nozadinske boje ćelija	iziaganja
		- veličina tablice	Metoda demonstracije
		- poravnanje teksta u ćelijama	
		- umetanje i brisanje redaka i	Metoda praktičnih
		stupaca	aktivnosti
		- spajanje i dijeljenje ćelija u	
		tablici te dijeljenje tablice	
Završni dio sata	5	Ponavljanje gradiva usvojenog na	Frontalna nastava
		današnjem satu uz dovršavanje zadatka	Individualni rad
		Uvod u sljedeći sat:	ivietoda razgovora
		- nretvaranje teksta u tablicu i	Metoda usmenog
		obratno	izlaganja
		- matematika u tablicama	Metoda praktičnih
			aktivnosti

# Tijek izvođenja nastave – nastavni rad (mikroplan) Uvodni dio

U uvodnom dijelu sata ću se predstaviti učenicima, iznijet ću im i utvrditi pravila i način izvršavanja sata, upoznat ću ih sa sadržajem današnjeg sata te ću utvrditi predznanje koje imaju.

N: Dobro jutro!

U: Dobro jutro!

N: Moje ime je X.X i danas ću biti vaša profesorica. Prije nego što krenemo sa radom bih vas molila da saslušate par pravila. Ako netko ima pitanje ili želi odgovoriti na moje pitanje molim vas da prvo digne ruku. Molim vas nemojte odgovarati i pitati bez dizanja ruke. Ako možda ne primijetim odmah ruku u zraku, diskretno me pozovite, ili ako pak odlučim da sam usred misli i da nije pametno da prekidam, dat ću vam znak da sam vas vidjela i čim završim ću vas prozvati. Računala koristite da biste odradili svoj današnji zadatak. Paralelno rješavajući zadatak ćemo proći opcije koje Microsoft Word nudi u radu sa tablicama. Radit ćemo Microsoft Word i tablice, odnosno, umetanje i rad s tablicama. Rješavajući zadatak ćete naučiti oblikovati tablice što vam kasnije može biti od velike koristi – zbog toga bi bilo dobro da ga riješite. Odmah na početku rada ću vam pokazati koji je cilj, odnosno, rješenje našeg zadatka. Naravno, ako imate bilo kakvih pitanja, u bilo kojem trenutku dignite ruku i pitajte.

N: Što ste do sad radili u Microsoft Wordu?

U: (ovisno o tome koja su predznanja, učenici odgovaraju nakon dizanja ruke)

N: Dobro. Sva predznanja koja imate sa radom u Wordu vam mogu samo pomoći, ne i odmoći. Vidjet ćete kasnije na nekim primjerima kako možete iskoristiti ta prethodna znanja u oblikovanju recimo teksta unutar ćelija tablice ili oblikovanju samih ćelija. Ako malo zapnete, probajte se sjetiti što znate od prije i kako vam to znanje može pomoći u riješavanju novoga, odnosno, kako možete sa starim znanjem riješiti problem nastao u tablicama.

N: Zadatak ćemo raditi paralelno. Ako imate pitanja, pitajte, ako treba nešto ponoviti, recite. Ovako treba izgledati rješenje danas – zadatak je vrlo jednostavan i može se brzo riješiti, a sadrži sve segmente gradiva koje ćemo proći. N: Za početak ćemo otvoriti Microsoft Word – kratica se nalazi ili na Desktopu, taskbaru ili odite na Start pa ga potražite na popisu programa. Kad otvorite Word, otvorit će vam se novi dokument. Tko nije otvorio novi dokument ili ne može pronaći, neka digne ruku.

### 1.1.1. Glavni dio

U glavnom dijelu sata ću učenicima prezentirati kako se tablice kreiraju i oblikuju u Microsoft Wordu. Upoznat ću učenike sa sučeljem koje je vezano uz rad sa tablicama te ćemo paralelno rješavati zajednički zadatak podijeljen u manje cjeline.

N: Krajnje rješenje izgleda ovako – dvije su tablice koje ćemo stvoriti iz jedne. Prva tablica ima dva retka, druga ima deset redaka. U tablici ima i pozadinskih boja, boje teksta, spojenih ćelija i podijeljenih ćelija. Vidjet ćete kako ćemo po koracima polako dolaziti do svog rješenja.

N: Prvi korak u rješavanju našeg zadatka je samo kreiranje tablica. Tablicu možemo kreirati na više načina. Svima je zajedničko što ćemo koristiti istu karticu i isti alat – Umetni (Insert) i Tablica (Table) – samo ćemo koristiti drugu opciju. Tko nije pronašao tu karticu i alat, molim neka digne ruku.

N: Kada kliknete na alat Tablica (Table), vidjet ćete te opcije. Prva opcija je matrica kojom ćemo odrediti koliko stupaca i redaka želimo. Maksimalni broj stupaca je deset, a maksimalni broj redaka je osam. Također možete vidjeti koliko stupaca i koliko redaka ste odabrali, tako da ne trebate brojati kućicu po kućicu. Kada smo odredili koliko stupaca i redaka želimo, lijevim klikom miša potvrdite.

N: Drugi način kreiranja tablice je koristeći naredbu Umetanje tablice (Insert Table). Klikom na tu naredbu, otvara se dijaloški okvir u kojem određujemo broj stupaca i broj redaka. U ovom dijaloškom okviru možemo odrediti i širinu stupaca – automatski se postavi na opciju da je širina fiksna, ali možete je odrediti i širinom sadržaja, ili da se rasporedi automatski po cijelom dokumentu. Mi ćemo koristiti fiksnu širinu jer želimo da nam svi stupci budu jednako široki. Kliknemo na "OK" i kreirali smo novu tablicu.

N: Koristeći ovaj drugi način kreiranja tablice ćemo početi rješavati svoj zadatak – kreirat ćemo tablicu sa sedam stupaca i dvadeset redaka.

N: Ako vam se dogodi da se trebate vratiti korak unatrag, kako ćete to napraviti?

U: Koristeći opciju Undo (Poništi).

N: Kako ćete je koristiti?

U: Klikom na strelicu u alatnoj traci ili kraticom ctrl + Z.

- 95 -

N: A kako ćete otići korak unaprijed?

U: Koristeći opciju Redo (Ponovi) koja se nalazi pokraj Undo (Poništi) opcije, a kratica je ctrl + Y.

N: Tako je – ctrl + Z za Undo (Poništi) ili strelica "unatrag" i ctrl + Y za Redo (ponovi) ili strelica "unaprijed". Ima li netko tko nije gotov sa kreiranjem tablice?

N: Sad kad smo kreirali tablicu, možemo je početi oblikovati. Primijetite kako se kad kreiramo tablicu, odnosno kad je označimo, na alatnoj traci pojavljuju Alati tablice (Table Tools) i kartice koje omogućuju oblikovanje i rad s tablicom, Dizajn (Design) i Izgled (Layout).

N: S alatima u kartici Dizajn (Design) možete mijenjati oblik i izgled tablice, obrube, debljinu i izgled obruba, možete mijenjati pozadinsku boju. Tu su i stilovi kako tablica može izgledati. S alatima u kartici Izgled (Layout) ćemo se baviti detaljnije dok ćemo rješavati zadatak pa ću vam pojasniti što radi koja grupa alata.

N: Naravno, svaki put kada želite promijeniti tekst, ćeliju, redak i stupac, trebate se prvo pozicionirati i odabrati ono što želite promijeniti. Zapamtite da iz svakog alata izlazite sa ESC ili ponovnim klikom na taj alat.

N: Mi ćemo sad u svojoj tablici drugom retku promijeniti pozadinsku boju u sivu. Kako se mijenja pozadinska boja u Wordu?

U: Koristeći alat Sjenčanje (Shading).

N: Tako je – sječanje koje na traci možete prepoznati kao kanticu iz koje se izlijeva boja. Znači, ovaj zadatak ćemo riješiti tako da koristimo alat Sjenčanje (Shading).

N: No, da bismo riješili ovaj drugi zadatak, prvo moramo naučiti kako se označavaju retci, stupci, ćelije i cijela tablica.

N: Da bismo označili redak, pokazivačem dođemo sa lijeve strane tablice dok nam se pokazivač ne okrene zrcalno. Kad se okrene, kliknemo i označit ćemo cijeli red. Slično je sa stupcima – kad dođemo s gornje strane tablice, pokazivač će se okrenuti i zacrniti i strelica će gledati prema dolje. Kliknemo i označili smo cijeli redak. Ako želimo označiti ćeliju, trebamo doći u donji lijevi rub ćelije dok nam se strelica ne okrene i zacrni – ta strelica će gledati u gornji desni rub ćelije – klikom označujemo tu ćeliju. Cijelu tablicu označimo tako da pokazivačem koji se promijeni u križić sa strelicama na kraju kliknemo na kvadratić u gornjem lijevom kutu tablice koji također ima križić sa strelicama u sebi. N: Sad kada znamo kako se mijenja pozadinska boja i kako možemo označiti cijeli red, lako možemo odraditi svoj zadatak.

N: Naš sljedeći zadatak je brisanje donjeg obruba na neparnim retcima brojeći od sivog retka. Dakle, prvi redak ispod sivog retka je broj jedan i od njega brojimo sve neparne. Kako ćemo označiti istovremeno više redaka, stupaca ili ćelija koji su međusobno odvojeni?

U: Koristeći ctrl.

N: Tako je – ctrl. Onda koristeći to znanje, označite neparne retke i maknite im donji obrub. Na kraju ovog zadatka, tablica će izgledati ovako. Tko nije gotov ili ima pitanja, neka digne ruku.

N: Sad ćemo se pozabaviti malo veličinom tablice. Veličinu tablice, stupaca i redaka možemo promijeniti na dva načina. Ako želimo promijeniti veličinu cijele tablice, postavit ćemo se pokazivačem u donji desni kut tablice gdje možete primijetiti mali kvadratić. Kada vam se pokazivač promijeni u obostranu strelicu, kliknite lijevom tipkom miša. Pokazivač će vam se ponovno promijeniti, ovaj put u križić, a veličinu možete mijenjati po želji povlačeći u smjeru u kojem vam treba. Primijetite kako vam se stvaraju ove crtkane linije koje pokazuju koliko bi vam tablica bila velika da sad pustite tipku miša. Kako možemo postići da nam se širina i visina tablice mijenjaju proporcionalno?

U: Koristeći Shift.

N: Tako je – koristeći Shift. Znači, ako želimo da nam se veličina tablice mijenja proporcionalno, dok povlačimo tablicu po dokumentu, neprestano držeći lijevu tipku miša, držat ćemo i Shift.

N: Dalje, ako želimo promijeniti veličinu stupca ili retka, postavimo se na obrub stupca ili retka dok nam se pokazivač ne promijeni u dvije paralelne linije sa strelicama koje idu od nje. Ako je u pitanju stupac, taj novi pokazivač će biti okomit, a ako je u pitanju redak, horizontalan. Kad dobijete taj pokazivač, lijevom tipkom miša kliknete i povlačite gore-dolje, odnosno lijevo-desno, dok ne dobijete željenu visinu, odnosno širinu.

N: Mi ćemo u svojoj tablici promijeniti visinu parnih redaka, ponovno brojeći od sivog retka, na 2cm. Označit ćemo ih sve istovremeno, i koristeći na kartici Izgled (Layout) u grupi Veličina ćelije (Cell Size) ili naredbama u dijaloškom okviru Svojstva tablice (Table Properties) ćemo upisati željenu vrijednost visine redaka. Dijaloški okvir se otvara ako kliknemo na strelicu u donjem desnom kutu grupe Veličina ćelije (Cell Size). Kada smo upisali željenu vrijednost, stisnemo Enter ili kliknemo na "OK". Na kraju bi vam vaša tablica trebala izgledati kako je prikazano. Tko nije uspio ili ima pitanja, neka digne ruku.

- 97 -

N: Tekst u tablici možemo urediti istim alatima kao i tekst izvan tablice. Isto je i sa poravnanjima. Mi ćemo sad u svoju tablicu upisati malo teksta. U prvi redak – ovaj put je prvi redak onaj iznad sivog retka – ćemo upisati "Svibanj". U sivi redak ćemo upisati redom dane u tjednu, a nakon toga ćemo u neparne retke, u svaku ćeliju upisati brojeve od 1 do 31. Brojeve ćemo počet upisivati tako da nam je broj "1" ispod četvrtka. Ima li netko tko nije upisao sav tekst?

N: Nakon što smo upisali tekst, malo ćemo ga oblikovati. "Svibanj" ćemo povećati na 15 i podebljat ćemo ga. Dane u tjednu ćemo podebljati i centrirati, a brojeve ćemo samo podebljati.

N: Koje su kratice za podebljavanje teksta, potcrtavanje teksta i kurziv?

U: ctrl + B, ctrl + U, ctrl + I

N: Tako je – Bold, Underline, Italic.

N: Retku sa danima u tjednu ćemo promijeniti visinu na 1cm i tekst poravnati u sredinu i u sredinu – vertikalno i horizontalno. Dakle, označite taj redak i poravnajte tekst koristeći grupu naredbi Poravnanje (Alignment). U ovoj grupi možete primijetiti da tekst možete poravnati lijevo, u sredinu ili desno (left-center-right) horizontalno, odnosno, gore, u sredinu i dolje (top-middle-bottom) vertikalno. Također ovdje postoji alat za okretanje teksta, odnosno, mijenjanje smjera teksta te za prilagođavanje margina ćelija, odnosno, udaljenosti teksta od ruba ćelije. Ima li netko tko nije napisao i uredio sav tekst?

N: Sad ćemo naučiti umetanje i brisanje redaka i stupaca. Ako želimo obrisati ili dodati redak ili stupac, koristit ćemo grupu naredbi Retci i stupci (Rows & Columns) na kartici Izgled (Layout). Retke možemo dodavati iznad ili ispod označenog retka, a stupce lijevo ili desno od označenog. Koliko redaka ili stupaca odaberete, toliko ćete ih i stvoriti – znači, ako želim da mi se stvore dva nova retka, označit ću dva nova retka i odabrati opciju za umetanje redaka (iznad ili ispod). Ako pak želimo obrisati retke ili stupce, koliko ih želimo obrisati, toliko ih i odaberemo i obrišemo. Primijetite kako je moguće brisati i ćelije i cijelu tablicu. Zna li netko koji je još način za brisanje redaka, ćelija ili stupaca?

U: Desni klik na ćeliju i odaberemo Delete Cells, gdje onda odaberemo hoćemo li obrisati ćeliju, redak ili stupac.

N: Tako je. Ponekad je to brži način za brisanje redaka. Mi ćemo u svojoj tablici obrisati suvišne retke – zadnjih 8. Pripazite – iako izgleda da je samo četiri retka, osam ih je jer smo obrisali obrube. Naša tablica sad već bolje izgleda – tko nije došao do ovog izgleda tablice, molim neka digne ruku. N: Word je jako pregledan i vrlo je lagano urediti dokument. Može li mi netko reći koji OpenSource program koji je ekvivalent Wordu?

U: LibreOffice ili OpenOffice Writer.

N: Tako je – Writer je vrlo poznati primjer. Ti programi su vrlo dobri i u njima možete napisati također kvalitetne radove. Malo je možda drugačije sučelje i ponekad je potrebno vremena da se navikne, ali, u svakom slučaju probajte da vidite da postoje i drugi programi osim Worda.

N: Danas ćemo još naučiti spajanje i dijeljenje ćelija u tablici. Ćelije možemo spojiti na dva načina – prvi način je ako odaberemo naredbu Spoji ćelije (Merge Cells) na kartici Izgled (Layout), a drugi je način ako kliknemo desni klik na mišu i zatim odaberemo Spoji ćelije (Merge Cells). Koji god način odabrali prvo trebate odabrati ćelije koje želite spojiti.

N: Mi ćemo spojiti svoje ćelije iz trećeg i četvrtog retka te trećeg stupca. Ako malo bolje pogledate, to su nam "prazne" kućice koje nemaju nikakav broj i zapravo nam niti ne trebaju. Kada označimo ćelije koje želimo spojiti – desni klik i Spoji ćelije (Merge Cells). Ako želimo da izgleda malo urednije, toj novoj ćeliji ćemo obrisati lijevi obrub. Kako ćemo to napraviti?

U: Pozicionirat ćemo se u tu ćeliju i u Dizajnu (Design) obrisati lijevi obrub.

N: Tako je – u kartici Dizajn (Design), pronađemo Obrubi (Borders) i maknemo lijevi obrub. Imamo još jednu kućicu koja ima "višak" obruba. Koja je to?

U: U donjem desnom kutu, zadnja dva retka i zadnji stupac.

N: Tako je – i tu ćemo na isti način obrisati obrube – desne i donji. Još ćemo par ćelija spojiti – one u prvom retku. Kako ćemo to napraviti?

U: Označit ćemo svih sedam ćelija, desni klik i zatim odaberemo Spoji ćelije (Merge Cells) ili u grupi alata Spoji ćelije (Merge Cells) odaberemo opciju za spajanje.

N: Tako je – odaberemo sve ćelije koje želimo spojiti i spojimo ih s opcijom Spoji ćelije (Merge Cells). U ovom retku ćemo naučiti i koristiti dijeljenje ćelije. Pozicionirajte se u taj redak i odaberite naredbu Podjela ćelija (Split Cells). Otvara nam se dijaloški okvir Podjela ćelija (Split Cells) gdje određujemo na koliko će se dijelova ćelija podijeliti. Mi ćemo podijeliti redak u dva stupca – ostavite dakle jedan redak, ali dodajte da budu dva stupca. Nove ćelije uredite tako da im tekst bude na sredini, a u drugu ćeliju koju smo stvorili upišite "2014.", podebljajte i povećajte na 15. N: Na kraju ćemo još samo podijeliti tablicu u dvije i to tako što ćemo je prelomiti ispod retka sa danima u tjednu koristeći opciju Podijeli tablicu (Split Table) – tablica će se podijeliti iznad retka koji ste odabrali tako da biste na kraju trebali dobiti dvije tablice. U drugoj tablici ćete primijetiti da nam je ostao obrub, pa ćemo ga obrisati da bude urednije.

N: Na kraju ćemo dobiti jednu tablicu sa dva retka u kojoj imamo upisan mjesec i godinu te dane u tjednu i tu tablicu kasnije možemo samo "kopirati" ako želimo napraviti kalendar za cijelu godinu, i jednu tablicu sa danima u tjednu koju je potrebno modificirati za svaki mjesec posebno. Mi bismo u svojoj tablici trebali još označiti neradne dane – brojeve ćemo pobojati u crveno. Kako mijenjamo boju teksta?

U: Koristeći Font Color.

N: Tako je – koristeći Font Color. U svibnju je neradni dan samo 1. i nedjelje. Tko nije uspio ili nije gotov, molim neka digne ruku. Naša tablica bi trebala izgledati ovako. U parne retke, ako si nećete printati kalendar, upisujete svoje obaveze, npr. 1. svibnja je Praznik rada.

### 1.1.2. Završni dio

U završnom dijelu sata ću ponoviti s učenicima što smo sve prošli te ću pregledati po razredu jesu li svi došli do krajnjeg rješenja. Na taj način ću provjeriti jesu li učenici shvatili današnje gradivo te ću po potrebi ponovno proći dijelove za koje primijetim da učenici imaju više poteškoća.

N: Kako možemo kreirati tablice?

U: Prvi način je da nakon što kliknemo na karticu Umetni (Insert), kliknemo alat Tablica (Table) i povlačenjem miša određujemo broj redaka i stupaca. Drugi način kreiranja tablica je naredbom Umetanje tablice (Insert Table) tako da upisom u dijaloški okvir odredimo broj redaka i stupaca.

N: Kako ćemo promijeniti pozadinsku boju, a kako boju teksta u tablici?

U: Na isti način kao i tekst izvan tablice – koristeći Font Color i Shading.

N: Kako sve možete promijeniti veličinu stupaca i redaka?

U: Postavimo pokazivač miša uz rub stupca ili retka i kada poprimi oblik dvije okomite crte sa strelicama koje idu od njih, povlačimo ga u određenom smjeru ovisno o tome želimo li stupac ili redak proširiti ili suziti. Ili naredbama na kartici Izgled (Layout) u grupi Veličina ćelije (Cell Size) ili naredbama u dijaloškom okviru Svojstva tablice (Table Properties). N: Koje naredbe koristimo za spajanje, a koje za dijeljenje ćelija?

U: Spoji ćelije (Merge Cells) na kartici Izgled (Layout) odnosno Podjela ćelija (Split Cells) na kartici Izgled (Layout).

N: Kako ćemo podijeliti jednu tablicu u dvije?

U: Koristeći naredbu Podjela tablice (Split Table).

N: U radu sa tablicama postoje još dva elementa koja je ćete naučiti, a to su matematika i pretvaranje teksta u tablicu i obratno. Word može u tablicama izvoditi jednostavne matematičke operacije (npr. zbrajanje, množenje, određivanje najveće vrijednosti i sl.) i to pomoću alata tablice Formula (Formula) na kartici Izgled (Layout). Na vrlo jednostavan način možete pretvoriti tekst odvojen razdjeljnikom (zarezom, točkom, razmakom...) u tablicu koristeći Umetni (Insert) > Tablica (Table) > Pretvaranje teksta u tablicu (Convert Text to Table). Isto tako, ponekad je potrebno tablicu pretvoriti u tekst što je također vrlo jednostavno za napraviti – označimo tablicu, stupac ili redak i u kartici Izgled (Layout) pronađemo naredbu Podatci (Data) te odaberemo Pretvaranje u tekst (Convert to Text). I kod pretvaranja teksta u tablicu i kod pretvaranja tablice u tekst će nam se otvoriti dijaloški okvir u kojem onda biramo dodatne opcije.

N: Ta dva elementa ćete naučiti sljedeći put, a matematiku u tablicama ćete učiti i kad ćete početi raditi u Excellu. Imate li kakvih pitanja vezanih uz današnje gradivo?

N: Vaša domaća zadaća je da napravite kalendar do kraja – imate svibanj, dakle, ostaje vam još 7 mjeseci do kraja godine. Pripazite na dane u tjednu i neradne dane.

N: Uskoro će zvoniti, pa vas molim da prije izlaska iz razreda ugasite računala i poklopite ih i lijepo spremite svoje radno mjesto. Molim vas nemojte ostaviti smeće na stolu i oko stola nego lijepo sve počistite. Hvala! Doviđenja i ugodan dan vam želim!

U: Doviđenja!

#### Kraj pripreme

### 2.2. Primjer pisane pripreme 2. –individualno predavanje

Priprema je izrađena za drugi razred usmjerenja opća gimnazija za učenje programiranja odnosno sortiranja u programskom jeziku C++.

#### Početak pripreme:

Škola: Prva gimnazija, Varaždin

Razred: 2.razred

Nadnevak izvođenja nastavnog sata: 19. svibnja 2014.

Nastavna jedinica: Sortiranje jednodimenzionalnih polja (nizova)

Nastavna cjelina: Sortiranje razmjenom (eng.Selection sort)

Tip sata: obrada novog gradiva

#### Cilj nastavnog sata:

Upoznati učenike sa sortiranjem jednodimenzionalnih polja Sortiranjem razmjenom (Selection sort).

#### Materijalni zadaci:

- Kombinirati ostalu funkcionalnost C jezika u radu s jednodimenzionalnim poljima
- Primijeniti mogućnosti sortiranja razmjenom nad jednodimenzionalnim poljima
- Izraditi programske zadatke upotrebom sortiranja razmjenom.

#### Funkcionalni zadaci:

• Razvijati dobru orijentaciju unutar okruženja C jezika

- Osposobljavati učenike za razumijevanje praktične primjene algoritma sortiranja razmjenom
- Poticati kreativan pristup rješavanju konkretnog zadatka u C jeziku promjenom algoritma sortiranja razmjenom
- Poticati interes za samostalno otkrivanje ostalih mogućnosti primjene algoritma sortiranja razmjenom

#### Odgojni zadaci:

- Razvijati smisao za red i organizaciju sadržaja
- Stjecati pozitivan stav prema radu
- Razvijati suradnju i međusobno pomaganje

#### Nastavna pomagala:

- Računalo
- Projektor

#### Nastavne metode:

- Metoda razgovora
- Metoda usmenog izlaganja
- Metoda demonstracije
- Metoda praktičnog rada

### Oblici rada:

- Frontalni rad
- Samostani rad

### Literatura:

- Tatjana Stranjak, Vesna Tomić, C-jezik, udžbenik za drugi i treći razred srednje škole, Zagreb,
   2005. Školska knjiga
- Nina Lipljin, Programiranje 1, Varaždin, 2004. Tiva Tiskara
- B. Motik, J.Stribar, Demistificirani C++, 2010., Element

Koraci nastavnog procesa i trajanje u minutama		Sadržaj koraka
Uvodni dio sata	5 minuta	Pozdrav i predstavljanje
		Upoznavanje s pravilima i načinom rada
		Uvodna pitanja
		Upoznavanje učenika sa sadržajem današnjeg
		nastavnog sata

Glavni dio sata 1. sat	35 minuta	Upoznavanje s pojmom sortiranja
		Upoznavanje s algoritom sortiranja
		razmjenom
		Vizualne demostracije algoritma sortiranja
		razmjenom
		Demonstracija i praktičan rad algoritmom
		sortiranja razmjenom
		Rezime
Završni dio sata	5 minuta	
		Ponavljanje
		Uputa za kraj sata
		Pozdrav

#### TIJEK IZVOĐENJA NASTAVE

#### Uvod (5 minuta)

N: Dobar dan! Moje je ime X.X i danas ću biti vaš nastavnik. Molio bih vas da tijekom rada komuniciramo na način da podignete ruke i tek kada budete prozvani postavite svoje pitanje ili dajte adekvatan odgovor. Na vrijeme ćete biti obaviješteni kada dođe vrijeme odmora.

N: Obavijestit ću vas kada ćete uključiti računala.

(Ukoliko su učenici već uključili računala, tražim od njih da zaklope računala.)

N: Na današnjem satu ćemo upoznati mogućnosti sortiranja te iste primijeniti u programiranju nad jednodimenzionalnim poljima. Upoznat ćemo sortiranje razmjenom kao najjednostavniji algoritam sortiranja jednodimenzionalnih polja, a ujedno ćemo primijeniti i dosadašnja znanja iz jednodimenzionalnih polja.

#### ASOCIJACIJSKA PITANJA VEZANA ZA SORTIRANJE :

N:U gimnazijskom programu imate nastavu iz predmeta povijest. Nad čime bismo u tom predmetu mogli primijeniti sortiranje ?

U: Sortiranje bismo mogli primijeniti nad događajima koji su se dogodili.

N: Tako je. Nad događajima možemo primijeniti sortiranje.

N: Prema čemu sortiramo događaje u povijesti?

U: Najčešće ih sortiramo prema godinama.

N: Odlično. Slijed godina u povijesti je jako dobar primjer sortiranja. Godine su sortirane od prve godine odnosno od najmanje godine do najveće.

N: Da li u knjigama postoji sortiranje?

U: Postoji.

N: Sortiranje postoji. A što se to sortira u knjigama?

U: Primjerice broj stranica.

N: Da. Stranice u knjigama predstavljaju uređen i sortiran niz.

N: Zamislite samo kako bi se bilo teško snalaziti u knjizi kada stranice ne bi išle po nekom redu.

N:Što još iz svakodnevnog života možete navesti kao sortiran niz, a bitan vam je da biste znali koji je dan u godini?

U: Kalendar može biti sortiran niz.

N: Da, kalendar je isto sortiran i uređen niz.

#### Glavni dio sata: (35 minuta)

N:Kada je riječ o bilo kakvom sortiranju, prve asocijacije su uvijek slaganje prema veličini ili nekom određenom parametru. Recimo knjige u knjižnici su sortirane prema abecedi ili prema nazivu autora ili prema području.

#### (Krećem s prezentacijom "Sortiranje elemenata niza")

N:Sortiranje niza isto tako predstavlja slaganje po veličini.

N: Od čega se sastoji polje?

U: Od elemenata polja ili varijabli.

N:Dobar odgovor. Dakle u poljima ćemo sortirati elemente po veličini, a ne indekse. To je vrlo važno za zapamtiti.

N: To znači da je sortiranje niza postupak kojim se elementi niza slažu prema veličini. Postoji više algoritama sortiranja i razlikuju se po složenosti i efikasnosti. Neki algoritmi su vrlo jednostavni i primjenjuju se za sortiranje niza s manjim brojem elemenata, dok su drugi namijenjeni za sortiranje većeg broja elemenata. Isto tako efikasnost se odnosi na to koliko brzo i točno algoritam sortira neki niz prema veličini.

N: Razlikujemo: jednostavnije algoritme sortiranja nizova koji su dobri za sortiranje nizova s manjim brojem elemenata (nekoliko stotina elemenata) te složenije algoritme sortiranja nizova koji su dobri za sortiranje nizova s većim brojem elemenata (nekoliko tisuća elemenata).

N: Algoritam Sortiranja razmjenom ili engleski Selection sort je jedan od najjednostavnijih algoritama sortiranja. Vrlo je efikasan za nizove s oko petsto elemenata.

N: Svaki algoritam ima korake koje provodimo da bismo došli do određenog rješenja. Stoga algoritam Sortiranja razmjenom ima tri koraka.

N: U prvom koraku na prvo mjesto se dovodi najmanji broj, u drugom koraku na drugo mjesto dovodi se najmanji od preostalih. Isti postupak se ponavlja do kraja dok se ne dobije sortirani niz.

N: Dakle, sortiranje razmjenom podrazumijeva da se vrijednost svakog elementa niza, počevši od prvog, uspoređuje se s vrijednostima elemenata koji dolaze iza njega. Ako je poredak dvaju elementa koja se uspoređuju obratan od redoslijeda sortiranja, elementima se zamijene vrijednosti.

N: Zamjenu vrijednosti je moguće napraviti na više načina. Dva najpraktičnija načina su: razmjena putem računskih operacija zbrajanja i oduzimanja i razmjena dodavanjem pomoćne varijable.

N: Prvi način (razmjena putem računskih operacija zbrajanja i oduzimanja) izgleda ovako:

x=6, y=4


N: Umjesto ovih brojeva mogu biti bilo koji drugi brojevi. Postupak je potpuno isti za bilo koji drugi broj.

N: Dakle, postupak razmjene kreće na način da prvo zbrojimo vrijednosti varijable x i y i tu novu vrijednost upišemo u varijablu x. U drugom koraku u varijablu y upisujemo razliku vrijednosti varijable x i varijable y. U završnom koraku u varijablu x zapisujemo novu vrijednost koju smo dobili razlikom vrijednosti varijable x i y.

N: Ako bismo sada htjeli u C-u ispisati vrijednost, koju varijablu bi uključili u ispis: x ili y?

U: Upisali bi varijablu x.

N: Tako je. Upisali bi varijablu x zato jer je u njoj zapisana posljednja vrijednost.

N: Sada ćemo na konkretnom nizu vidjeti kako algoritam sortiranja razmjenom upotrebom operacija zbrajanja i oduzimanja sortira niz prema veličini.

N: Prisjetimo se da općenito u informatici velik broj operacija je riješen vrlo jednostavnim pristupom odnosno operacijom zbrajanja i oduzimanja. Primjerice operacija dvostrukog komplementa gdje se koristi jednostavna operacija zbrajanja. Sad vidite odakle dolazi ideja za sortiranje upotrebom operacija zbrajanja i oduzimanja.

N:U početku imamo nesortiran niz čije vrijednosti nisu poredane po veličini. Početak izvođenja algoritma započinje pozicioniranjem na početnu vrijednost niza. U ovom slučaju je to vrijednost 5.

(Provodim vizualne demonstracije nad prezentacijom "Sortiranje elemenata niza" nad slajdovima 9-31 i povremeno uključujem učenike u korake izvođenja algoritma na način da nekog prozovem ako je netko od učenika podigao ruku ili da svojevoljno odaberem učenika ako se nitko ne javlja.)

N: Ima li kakvih pitanja u vezi ovog algoritma? Kome nije jasan algoritam i njegovi koraci izvođenja neka podigne ruku!

(Ukoliko nekolicini učenika ili pojedinom učeniku algoritam nije jasan, pokušavam objasniti vračajući se na slajd 7 i 8 to jest početak rada algoritma; ukoliko je svima jasno krećem dalje na iduću inačicu algoritma)

N: Drugi način razmjene elemenata ne uključuje operacije zbrajanja i oduzimanja, jer prvotni način iako efikasan, podosta je kompleksan i oduzima puno vremena.

N: Zamislite da želite 500 elemenata sortirati na najbrži mogući način jer vam se žuri ispostaviti uređene podatke, za izvođenje algoritma treba neko izvjesno vrijeme.

N: Upravo iz razloga da se vremenski skrati izvođenje algoritma, osmišljen je drugi način sortiranja uz pomoć dodatne varijable ili pomoćne varijable.

N: Zamislimo to ovako: Imate dvije limenke sa sokom. Jedna može biti Coca Cola, a druga Fanta. Želite se malo našaliti s nekim prijateljem i odlučili ste zamijeniti sadržaj limenki. Najbrži način da to učinite jest da jednu od limenki ispraznite u dodatnu posudu odnosno čašu te da nakon toga limenku sa sadržajem prelijete u praznu limenku i onda na kraju iz čaše vratite sadržaj u drugu praznu limenku. I na kraju ste dobili razmjenu sadržaja limenki.

N: Inačica algoritma s pomoćnom varijablom nalikuje upravo zamjeni sadržaja između dvije limenke soka.

(Uz objašnjavanje provodim vizualnu demonstraciju i povezujem to s algoritmom prolazeći slajdovima 33-38 prezentacije "Sortiranje elemenata niza")

N: U nekom pretkoraku svakako moramo otvoriti limenke soka.

N: Sada dodajemo čašu koja predstavlja pomoćnu varijablu u koju ćemo privremeno smjestiti sadržaj neke od limenki odnosno neke od varijabli.

N:Jednu od limenki istočimo u čašu odnosno ako je to varijabla x s nekom vrijednosti, vrijednost smo spremili u pomoćnu varijablu. Drugu (punu) limenku odnosno drugu varijablu y istočimo i spremimo sadržaj u prvu varijablu odnosno limenku. Naposljeku moramo vratiti sadržaj pomoćne varijable odnosno čaše u malo prije ispražnjenu limenku odnosno varijablu y. I na taj način napravili smo razmjenu.

N: Da vidimo kako to izgleda kada radimo s konkretnim varijablama, a ne limenkama i sokovima.

• Početne varijable x=6 i y=4

t=x; -> vrijednost varijable x smješta se u pomoćnu varijablu t

x=y;-> u varijablu x smješta se vrijednost varijable y, dakle x=4

t=y; -> u varijablu y smješta se vrijednost pomoćne varijable t, dakle y=6

Rezultat: x=4 i y=6

N: Početne varijable su vrijednosti x=6 i y=4. Već na samom početku dodajemo pomoćnu varijablu t. Primijetite u nju smještamo vrijednost varijable x, dakle vrijednost 6.

N:Kolika je sad vrijednost varijable x?

U: Vrijednost varijable x je sad 0.

N: Točan odgovor.

N: Iz varijable y smještamo vrijednost 4 u varijablu x.

N: Koja se sad vrijednost nalazi u varijabli x?

U: Vrijednost 4.

N:Tako je.

N: I to ćemo napraviti u posljednjem koraku?

U: Vrijednost pomoćne varijable t upisujemo u varijablu y.

N: Tako je. Koju smo vrijednost vratili u varijablu y?

- 110 -

U: Vrijednost 6.

N: Krajnje rješenje je x=4 i y=6.

N:Uočite koliko je ova inačica algoritma jednostavnija i brža za razliku od inačice s operacijama zbrajanja i oduzimanja.

N: Pogledajmo sada na konkretnom nizu.

N: Primjer niza je jednak kao i prethodni primjer kod inačice sortiranja uz operacije zbrajanja i oduzimanja.

N: Ove strelice crvena i plava sada dobivaju uloge. Crvena ima ulogu početka odnosno pozicioniranja na prvu varijablu niza, a plava to jest ona koja se kreće nizom predstavlja pomoćnu varijablu u koju se privremeno zapisuje vrijednost na koju pokazuje.

(Provodim vizualne demonstracije nad prezentacijom "Sortiranje elemenata niza" nad slajdovima 40-62 i povremeno uključujem učenike u korake izvođenja algoritma na način da nekog prozovem ako je netko od učenika podigao ruku ili da svojevoljno odaberem učenika ako se nitko ne javlja.)

N: Ovim primjerom, vidjeli smo korak po korak kako radi inačica algoritma sortiranja razmjenom uz upotrebu pomoćne varijable.

N: Ako ima netko kome nije jasan postupak neka podigne ruku.

(Ukoliko postoji netko da nije jasno postupak vraćam se na slajd 30-38 i objasnim postupak na najjednostavnijem primjeru izjednačavajući pritom varijable x i y s limenkama; u suprotnom krećem dalje s predavanjem)

N: Sada ćemo proći kroz sortiranje razmjenom upotrebom pomoćne varijable, ali na način kako to radi računalo. Uloge onih strelica (one plave i crvene) su iste kao i u prijašnjem primjeru jednostavnog niza.

(Provodim vizualne demonstracije pokrenuvši link na prezentaciji "Sortiranje elemenata niza" na slajdu 63 i povremeno uključujem učenike u korake izvođenja algoritma na način da nekog prozovem ako je netko od učenika podigao ruku ili da svojevoljno odaberem učenika ako se nitko ne javlja.) (Ukoliko postoji netko da nije jasno postupak vraćam se na slajd 30-38 i objasnim postupak na najjednostavnijem primjeru izjednačavajući povezujući limenke s trenutnim nizom; u suprotnom krećem dalje s predavanjem) N:Da vidimo kako u programskom kodu izgleda sortiranje razmjenom uz upotrebu pomoćne varijable.

1.	int i, j;
2.	for (i=0; i <n-1; i++)<="" td=""></n-1;>
3.	for(j=i+1; j <n; j++)<="" td=""></n;>
4.	if(x[j] <x[i])< td=""></x[i])<>
5.	{
6.	/*uporaba računskih operacija zbrajanja i oduzimanja*/
7.	
8.	pom=x[i];
9.	x[j]=x[j];
10.	x[j]=pom;
11.	}

N: Pokušajmo povezati uloge onih dviju strelica one plave i crvene s ove dvije for petlje.

N: Koju ulogu ima crvena strelica?

U: Ima ulogu pozicioniranja na početni element niza.

N: Bravo, tako je.

N: Da li se uloga crvene strelice odnosi na plavu ili zelenu petlju?

U: Uloga crvene strelice se odnosi na plavu for petlju.

N: Tako je. Plava for petlja je zapravo taj pokazivač na početni element polja.

N: Po kojim parametrima u for petlji to možemo zaključiti?

U: Inicijalizacija varijable i je postavljena na 0 i i<n-1.

N: Bravo. Kada je i inicijalizirano na indeks O znači da se nalazi na prvom elementu.

N: Sada možemo zaključiti da je zelena for petlja u ulozi pomoćne varijable.

N: No koji uvjeti u njoj ukazuju da će biti pomoćna varijabla?

U: Na to ukazuje inicijalizacija j=i+1 i j<n.

N: Tako je. Inicijalizacija varijable kao j=i+1 znači da će ta pomoćna varijabla ili ona plava strelica iz naših primjera biti jedno mjesto ispred pozicionirajuće varijable (crvene strelice).

N: Pogledajmo sada kako u programskom kodu izgleda sortiranje razmjenom uporabom računskih operacija zbrajanja i oduzimanja.

1.	int i, j;
2.	for (i=0; i <n-1; i++)<="" td=""></n-1;>
3.	for(j=i+1; j <n; j++)<="" td=""></n;>
4.	if(x[j] <x[i])< td=""></x[i])<>
5.	{
6.	/*uporaba računskih operacija zbrajanja i oduzimanja*/
7.	
8.	x[i]=x[i]+x[j];
9.	x[j]=x[i]-x[j];
10.	x[i]=x[i]-x[j];
11.	}

N: Primijetite da je prvi dio prilikom inicijalizacije i dvije for petlje zapravo jednak. Promjena je samo u ovom donjem dijelu gdje umjesto pomoćne varijable postoje operacije zbrajanja i oduzimanja.

N: Krenimo sada primijeniti ovaj algoritam na konkretnom zadatku.

N: Sada možete uključiti računala i pokrenuti okruženje za rad u C jeziku. Pogledajte gdje se nalazi okruženje za rad. Nekima se nalazi na radnoj površini. Ukoliko nije na radnoj površini, potražite u start->programs.

N: Prije nego isprobamo algoritam sortiranja moramo imati polje te njegov unos i ispis.

#### Zadatak 1 (trajanje: 5 minuta):

- Izradite program nana način da ima:
- 1. Deklaraciju polja od 10 elemenata a[10]
- 2. Izbornik kojim definirate broj elemenata u polju

printf ("upisite broj elemenata u polju:");

3. Uz pomoću if else ispituje uvjet da li je broj elemenata manji 10

if(n<10)

- 4. Ispisuje poruku o grešci ukoliko uvjet nije zadovoljen
- 5. Obavezno koristiti for petlje u realizaciji rješenja

(Paralelno dok učenici pišu program, prođem razredom provjeravajući njihov rad uz određene intervencije)

(Nakon pet minuta postavljam pitanje: Ako netko nije uspio neka podigne ruku .Ukoliko postoje oni koji nisu uspjeli,interveniram kod njih te nakon toga pokažem finalno rješenje na slajdu prezentacije.

U suprotnom samo prikažem rješenje na slajdu prezentacije "Sortiranje elemenata niza" i prelazim na sljedeći zadatak)

N:Dakle u prethodnom zadatku deklarirano je polje duljine 10 elemenata s napravljenim mogućnostima proizvoljnog unosa elemenata te ispis tih istih elemenata.

N: U sljedećem zadatku ćemo primijeniti sortiranje razmjenom nad elementima polja pomoću računskih operacija zbrajanja i oduzimanja.

#### Zadatak 2 (trajnje: 4 minute):

- 1. Napravite sortiranje elemenata polja pomoću računskih operacija zbrajanja i oduzimanja
- 2. uz uvjet da sortiranje bude uzlazno

(Učenicima dajem kao pomoć slajd 69 u kojem se nalazi gotov algoritam)

(Paralelno dok učenici pišu program, prođem razredom provjeravajući njihov rad uz određene intervencije)

(Nakon četiri minute postavljam pitanje: Ako netko nije uspio neka podigne ruku .Ukoliko postoje oni koji nisu uspjeli,interveniram kod njih te nakon toga pokažem finalno rješenje na slajdu prezentacije.U suprotnom samo prikažem rješenje na slajdu prezentacije "Sortiranje elemenata niza" i prelazim na sljedeći zadatak)

N: Još jedanput pokrenimo program.

(Istovremeno i ja pokrećem svoje programsko rješenje)

N:Usredotočite se na ispis sortiranog niza.

N: U kakvom je redoslijedu niz sortiran, kako su brojevi poredani jedan za drugim?

U: Brojevi su poredani od najmanjeg do najvećeg.

N: Tako je. Brojevi su poredani od najmanjeg do najvećeg što znači da je niz sortiran uzlazno

N: Što mislite o kojem dijelu programskog koda zavisi da li će niz biti sortiran uzlazno ili silazno?

U: Zavisi o uvjetovanju na mjestu gdje je if(x[j]<x[i]).

N: Dobro. A što znači da je x[j]<x[i])?

U: Ako je x[j] na manjem mjestu u polju od pozicionirajuće varijable x[i], znači da se ona kreće od manjeg mjesta odnosno mjesta s manjom vrijednosti do pozicionirajuće varijable.

N: Bravo. Odlično razumijevanje. Ako je x[j] na manjem mjestu od x[i] ondataj x[j] mora u rastućem redoslijedu se kretati prema x[i] koji se nalazi na većem mjestu.

N: Dakle dok je x[j]>x[i]), imamo obrnutu situaciju odnosno situaciju kada se x[j] nalazi na većoj poziciji polja od x[i] i onda se ona s te veće poziciji mora vraćati silazno prema x[i];

#### Zadatak 3(trajanje: 3 minute):

- Program iz zadatka 2 preoblikujte da omogućava silazno sortiranje pomoću
- (Pčau a kklihoo pkata cijač z birai japajšui gabozi razmja prođem razredom provjeravajući njihov rad uz određene intervencije)

(Nakon tri minute postavljam pitanje: Ako netko nije uspio neka podigne ruku .Ukoliko postoje oni koji nisu uspjeli,interveniram kod njih te nakon toga pokažem finalno rješenje na slajdu prezentacije.U suprotnom samo prikažem rješenje na slajdu prezentacije "Sortiranje elemenata niza" i prelazim na sljedeći zadatak)

N: Sad ćemo isprobati i drugi način sortiranja uz upotrebu pomoćne varijable.

#### Zadatak 4 (trajanje: 4 minute)

- Prepravite prethodni program te napravite sortiranje elemenata polja pomoću
- Pomoćne varijable

(Učenicima dajem kao pomoć slajd 75 u kojem se nalazi gotov algoritam)

(Paralelno dok učenici pišu program, prođem razredom provjeravajući njihov rad uz određene intervencije)

(Nakon četiri minute postavljam pitanje: Ako netko nije uspio neka podigne ruku .Ukoliko postoje oni koji nisu uspjeli,interveniram kod njih te nakon toga pokažem finalno rješenje na slajdu prezentacije.U suprotnom samo prikažem rješenje na slajdu prezentacije "Sortiranje elemenata niza" i prelazim na sljedeći zadatak)

N: Primjenom ove inačice istog algoritma, kako sortiran niz ispisuje program?

U: Program opet ispisuje uzlazno sortiran niz.

N: Dobro. Što i gdje bismo trebali promijeniti da bi ostvarili silazno sortiranje niza?

U: Trebamo promijeniti uvjetovanje umjesto x[j]<x[i] staviti x[j]>x[i].

N: Zašto?

U:Zato što će se tada x[j] nalaziti ispred pozicionirajuće varijable odnosno skroz pri desnoj strani niza i morat će se spuštati od najvećeg indeksa niza prema najmanjem na mjesto gdje se nalazi x[i].

N: Odličan dogovor. Na taj način dobivamo silazno sortiran niz.

(Zadatak 5 i 6 su zadaci u pripravi ukoliko ostane viška vremena)

#### Rezime od 2 minute:

N: Evo današnjim satom smo utvrdili značaj i upotrebljivost algoritma sortiranja.

N: Dakle, radili smo s algoritmom sortiranja Selection sort ili Sortiranje razmjenom, iskušali njegove obje inačice putem pomoćne varijable i pomoću računskih operacija zbrajanja i oduzimanja.

N: Testirali smo u programu ispise sortiranih nizova i uočili da o ishodu sortiranog niza zavisi uvjetovanje dviju varijabli: pozicionirajuće varijable i pomične varijable koja se kreće kroz niz te promjena znaka < u znak >.

N: Taj algoritam ćete najčešće koristiti kao pomoć kod organizacije sadržaja niza odnosno elemenata niza.

N: Ponovimo:

#### Završni dio sata (5 minuta)

#### PITANJA ZA PONAVLJANJE:

N: Kako definiramo sortiranje niza u programiranju?

U: Definiramo ga kao je postupak kojim se elementi niza slažu prema veličini.

N: Koje inačice algoritma sortiranja razmjenom postoje?

U:Postoje sortiranje razmjenom upotrebom pomoćne varijable i sortiranje razmjenom pomoću računskih operacija zbrajanja i oduzimanja.

- 116 -

N:Koje i koliko petlji koristimo u algoritmu sortiranja?

- U: Koristimo for petlje i to dvije.
- N: Kako možemo sortirati niz algoritmom sortiranja razmjenom?
- U: Možemo sortirati uzlazno ili silazno.
- N: O čemu ovisi da li će sortirani niz biti sortiran uzlazno ili silazno?
- U: Ovisi o postavljenom uvjetu u uvjetovanju if(x[j]<x[i]) ili f(x[j]>x[i])
- (U posljednjoj minuti)
- N: Upravo će zvoniti hvala na suradnji. Isključite računala i pospremite radna mjesta.
- N: Doviđenja.
- U: Doviđenja.
- Školsko zvono.
- Kraj pripreme

# 3. Javna predavanja-osnovne škole

Pisane pripreme za izvođenje nastavnog sata u osnovnoj školi također započinju javnim predavanjima. Pripreme su pisane za dvije osnovne škole: Druga osnovna Varaždin i Šesta osnovna škola Obrazovni programi obje škole su usklađeni prema programima za izvođenje nastave na osnovnim školama. Isto vrijedi i za informatiku. Valja naglasiti kako pisanje priprema za osnovne škole treba prilagoditi kognitivnim mogućnostima učenika te dobi. Naime radi se o učenicima od petog do osmog razreda osnovne škole. Svako obrazovno razdoblje zahtijeva prvenstveno prilagodbu sadržaja dobi i kognitivnim mogućnostima učenika. U narednim stranicama ovog poglavlja biti će predstavljneni scenariji četiriju iznimno kvalitetnih priprema za izvođenje nastavnog sata na Drugoj osnovnoj školi Varaždin te Šestoj osnovnoj školi Varaždin. Radi zaštite osoba imena i prezimena se ne spominju u pripremama.

# **3.1.** Primjer pisane pripreme **1**. – javno predavanje

Nastavni sat, izveden je na **Drugoj osnovnoj školi Varaždin** nad učenicima **šestog razreda**. Učenicima je održan sat na temu **obrade teksta u MS Wordu**.

#### Početak pripreme:

# Priprema za izvođenje nastavnog sata

Nastavr	nik		Datum	20.11.2014.	
Škola		II Osnovna škola Varaždin	Razred	6.d	
Nastavr	ni predmet	Informatika			
Nastavr	na cjelina	Obrada teksta – Microsoft Word 20	13		
Nastavr	na jedinica	Izrada i oblikovanje tablica, kretanje	e u njima		
Cilj nast	avne jedinice	Upoznavanje učenika s unosom i ob	likovanjem tablica u	Microsoft Word-u	
Tip nast	avnog sata	Obrada novog gradiva			
Zadaci	Obrazovni	Definirati što je tablica i koji su osno	ovni dijelovi tablice		
		Navesti, opisati i demonstrirati nači	ne umetanja tablica		
		Opisati i demonstrirati označavanje	ćelija, redaka, stupad	a te cijele tablice	
		Navesti, opisati i demonstrirati nači	ne postavljanja širine	stupaca, visine	
		redaka te dimenzija cijele tablice			
		Opisati i demonstrirati umetanje dodatnih redaka, stupaca i ćelija			
		Opisati i demonstrirati brisanje redaka, stupaca i ćelija			
	Funkcionalni	Razvijati sposobnost samostalnog rada u Microsoft Wordu i sličnim			
		programima			
		Razvijati vještine rada na računalu			
		Utvrditi vještine rada u alatima MS	Office paketa (stvara	nje novog	
		dokumenta, spremanje, oblikovanje sadržaja)			
	Odgojni	Poticati samostalnost			
		Poticati osjećaj za preglednost, red	i urednost		
		Poticati pozitivne stavove prema radu			

Nastavne metode	Metoda razgovora
	Metoda usmenog izlaganja
	Metoda demonstracije
	Metoda praktičnih aktivnosti
Socijalni oblici rada	Frontalna nastava
	Individualni rad
Nastavna sredstva i	Računalo
pomagala	Projektor

# Organizacija nastavnog rada – artikulacija metodičke jedinice (makroplan)

Koraci nasta	vnog	Sadržaj koraka	Socijalni obli	ci rada i nastavne
procesa i trajanje u			metode	
minutama				
Uvodni dio	10	Pozdravljanje i predstavljanje	Frontalna	Metoda razgovora
		Upoznavanje s pravilima i načinom rada	nastava	Metoda usmenog
		Ponavljanje predznanja iz Microsoft Worda		izlaganja
		Upoznavanje učenika sa sadržajem		
		današnjeg nastavnog sata		
Glavni dio 60		Obrada što je tablica i koji su osnovni	Frontalna	Metoda razgovora
		dijelovi tablice	nastava	Metoda usmenog
		Demonstracija umetanja tablice (zadatak)	Individualni	izlaganja
		Demonstracija označavanja ćelija, redaka,	rad	Metoda
		stupaca te cijele tablice		demonstracije
		Demonstracija postavljanja širine stupaca i		Metoda praktičnih
		visine redaka te promjena veličine cijele		aktivnosti
		tablice (zadatak)		
		Demonstracija umetanja redaka i stupaca		
		(zadatak)		
		Demonstracija umetanja ćelija (zadatak)		
		Demonstracija brisanja stupaca i redaka		
		(zadatak)		

		Demonstracija brisanja ćelija (zadatak) Demonstracija brisanja cijele tablice (zadatak)		
Završni dio	20	Ponavljanje gradiva usvojenog na	Frontalna	Metoda razgovora
		današnjem satu	nastava	Metoda usmenog
		Uvod u sljedeći sat kroz samostalni zadatak	Individualni	izlaganja
		učenika	rad	Metoda
		Pozdravljanje učenika		demonstracije
				Metoda praktičnih
				aktivnosti

# Tijek izvođenja nastave – nastavni rad (mikroplan)

# Uvodni dio

U uvodnom dijelu sata ću se predstaviti učenicima, iznijet ću im i utvrditi pravila i način izvršavanja sata, upoznat ću ih sa sadržajem današnjeg sata te ću utvrditi predznanje koje imaju (pritom ću pratiti odgovore učenika te svaki točan odgovor pohvaliti i ponoviti). Tokom ponavljanja ću koristiti dokument METOD2\_Matekovic\_Ana\_JP1\_pitanja\_odgovori.doc u kojem se nalaze sva pitanja i odgovori.

N: Dobro jutro!

U: Dobro jutro!

N: Moje ime je X.X. i danas ću biti vaša profesorica. Prije nego što krenemo sa radom jedno pravilo. Ako netko ima pitanje ili želi odgovoriti na moje pitanje molim da prvo digne ruku. Molim vas nemojte odgovarati i pitati bez dizanja ruke.

N: Danas ćemo kroz primjere naučiti kako se kreiraju tablice i koje opcije MS Word nudi u oblikovanju tablica. Tablice se mogu koristiti kod uređivanja teksta, slaganja teksta po abecednom redu, prikazivanja podataka ili brojčanih vrijednosti nad kojima je moguće izvoditi matematičke operacije. Tablice možete koristiti da napravite raspored sati u školi, kalendar u koji ćete upisivati kad pišete koji test, popis ocjena, raspored treninga i slično.

N: MS Word ste već radili u petom razredu. Sad ćemo ukratko ponoviti što ste već radili.

# Ponavljanje [7min]

Svaki odgovor sa zvjezdicom ću demonstrirati na sučelju MS Word-a (prethodno ću otvoriti novi dokument za ponavljanje).

N: Za što sve koristimo MS Word?

N: Unijeli smo tekst u novi dokument.

\*N: Gdje možemo promijeniti izgled slova?

\*N: Gdje možemo promijeniti veličinu slova?

\*N: Kako tekst možemo podebljati?

\*N: Kako tekst možemo nakositi?

\*N: Kako tekst možemo potcrtati?

\*N: Gdje možemo odabrati kako će nam tekst biti poravnat?

\*N: Gdje biramo boju teksta?

\*N: Gdje određujemo pozadinsku boju?

N: Odlično. Ovo što smo sad ponovili će nam trebati malo kasnije. Sad ćemo krenuti sa radom sa tablicama u MS Word-u. Ako imate pitanja, dignite ruku i pitajte. Za početak otvorite MS Word – prečac se nalazi ili na radnoj površini ili otiđite na Start pa ga potražite na popisu programa. Kada vam se otvori Word, odaberite prazni novi dokument.

Otvorit ću novi dokument. Kroz cijeli sat ću raditi (demonstrirati opcije) u tom dokumentu.

N: Tko nije otvorio novi dokument ili treba pomoć, neka digne ruku.

# Glavni dio

U glavnom dijelu sata ću učenicima objasniti što su tablice i koji su osnovni dijelovi tablice, a zatim ću im pokazati kako se tablice kreiraju i oblikuju u MS Word-u. Demonstrirat ću opcije koje MS Word nudi za oblikovanje tablica pokazujući sve na primjerima – na kraju svake cjeline će učenici samostalno riješiti zadatak. Tokom obrade i vođenih vježbi ću postavljati pitanja – pratit ću odgovore učenika te svaki točan odgovor pohvaliti i ponoviti. Koristit ću dokument METOD2\_Matekovic\_Ana\_JP1\_pitanja\_odgovori.doc u kojem se nalaze sva pitanja i odgovori.

# Tablica i osnovni dijelovi tablice [4min]

**N:** Microsoft Word pruža raznovrsne mogućnosti umetanja i oblikovanja tablica. Tablica je organizirani način prikaza podataka složenih u retke (horizontalne linije) i stupce (vertikalne linije). Podaci su odijeljeni jedni od drugih rešetkama (obrubima).

*(demonstracija – Tablica 1: Dok objašnjavam što su stupci, što su retci, a što ćelije, otvorit ću dokument METOD1\_Matekovic\_Ana\_JP1\_primjeri.doc i pokazati tablicu na prvoj stranici tog dokumenta.):* Na presjeku stupca i retka nalazi se ćelija. Ćelija je osnovni element tablice i može sadržavati tekst, brojeve, slike pa i druge tablice. Sadržaj svake ćelije možete oblikovati kao i sav ostali sadržaj – poravnati, bojati, uvlačiti, obrubljivati i slično, neovisno o drugim ćelijama. Dakle, svaka ćelija može biti oblikovana drugačije.

Tablica 1 Osnovni dijelovi tablice - stupci, retci, ćelije

N (demonstracija – Tablica 2: Dok objašnjavam zaglavlja redaka i stupaca, otvorit ću dokument METOD1\_Matekovic\_Ana\_JP1\_primjeri.doc i pokazati tablicu na drugoj stranici tog dokumenta.): Zbog jednostavnijeg snalaženja među podacima svaka tablica bi trebala imati zaglavlja redaka i stupaca. U zaglavljima su najčešće nazivi i redni brojevi podataka.

	Ponedjeljak	Uto	orak	Srijeda	Četvrtak	Petak
1.	Eng iz.	V	je	Hrv	Tzk	Pov
2.	Eng iz.	V	je	Hrv	Hrv	Mat
3.	Gla	Н	rv	Pri	Inf	Pri
4.	SRO	Po	ov.	Nje	Inf	Nje
5.	Mat	Teh	Lik	Mat	Mat	Nje
6.	Hrv	Teh	Lik	Tzk		Zem
7.						Zem

Tablica 2 Zaglavlja redaka i stupaca

**N:** U paketu primjenskih programa Microsoft Office postoji program namijenjen samo za rad s tablicama – koji je to program?

# Umetanje tablice [10min]

N: Prvi korak je umetanje tablice. Microsoft Word nudi nekoliko različitih načina umetanja ili stvaranja tablice. Koji ćete od njih rabiti, ovisi o tome kako volite raditi i koliko je tablica složena za izradu. Ovdje ćemo prikazati dvije mogućnosti, a na vama je da odaberete koju ćete i kada primijeniti.

# Umetanje tablice označavanjem broja redaka i stupaca buduće tablice mišem

N *(demonstracija: Kreirat ću tablicu kakvu i učenici kasnije trebaju kreirati.)*: Molim vas da svi poslušate i pogledate koji su koraci prvog načina. Prvi način je umetanje tablice označavanjem broja redaka i stupaca buduće tablice mišem.

- Postavi se točka unosa ili se označi mjesto u dokumentu gdje se želi umetnuti tablica.
- Na kartici Umetanje (Insert), u grupi Tablice (Tables) se klikne na gumb Tablica (Table).
- Klikne se na prvi kvadratić/ćeliju i neotpuštajući lijevu tipku miša, povlačenjem, se označi potreban broj redaka i stupaca. Izgled tablice se vidi odmah tijekom povlačenja miša. Označene ćelije jasno su istaknute bojom.
- Kada se odabere željeni broj stupaca i redaka, otpusti se lijeva tipka miša.

**N:** Vaš prvi zadatak je sljedeći: kreirajte tablicu koristeći označavanje broja redaka i stupaca buduće tablice mišem. Tablica treba imati 6 stupaca i 8 redaka. **[2min]** 

# Tablica 3 Rješenje zadatka 1

**N:** Ima li netko tko nije uspio napraviti tablicu? Ako ste svi završili, onda možemo proučiti još jedan način umetanja tablica.

#### Umetanje tablice dijaloškim okvirom Umetni tablicu (Insert Table...)

N *(demonstracija: Kreirat ću tablicu kakvu i učenici kasnije trebaju kreirati.)*: Ponovno molim da saslušate i pogledate koji su koraci ovog načina. Sljedeći način umetanja tablice je pomoću dijaloškog okvira Umetni tablicu (Insert Table...).

- Na kartici Umetanje (Insert), u grupi Tablice (Tables) klikne se na gumb Tablica (Table), a zatim na naredbu Umetni tablicu... (Insert Table...).
- Otvorit će se dijaloški okvir Umetanje tablice (Insert Table...) sa slijedećim svojstvima tablice:
  - O Broj stupaca (Number of columns) i Broj redaka (Number of rows) koji omogućuje upis broja redova i stupaca buduće tablice.
  - Fiksna širina stupca (Fixed column width). Ovom naredbom se postavlja da stupac ćelije uvijek bude jednake širine, neovisno o količini upisanog sadržaja. Ako je sadržaj predugačak, povećat će se visina retka.
  - O Samoprilagodi sadržaju (AutoFit to contents). Širina stupca će se prilagođivati veličini upisanog sadržaja.
  - O Samoprilagodi prozoru (AutoFit to window). Često je potrebno raditi sa smanjenom veličinom prozora. Umetnuta će tablica zauzeti prostor od lijevog do desnog ruba prozora neovisno o njegovoj veličini.
  - O Zapamti dimenzije za nove tablice (Remember dimensions for new tables). Kada se odabere ova opcija, svaka sljedeća tablica će imati ista svojstva kao i prva tablica koja je kreirana.

N: Vaš sljedeći zadatak je da kreirate tablicu koristeći dijaloški okvir Umetni tablicu (Insert Table...) i to ispod tablice koju ste već napravili. Tablica treba imati 6 stupaca i 8 redaka, a ostale opcije neka ostanu kako jesu. Na kraju biste trebali imati svi dvije tablice u svojem dokumentu. [3min]

#### Tablica 4 Rješenje zadatka 2

N: Ima li netko tko nije uspio napraviti tablicu na ovaj način?

(Provjeravam učenike jesu li završili zadatke 1 i 2.)

## N: Koliko ćelija ima naša tablica?

N *(demonstracija na sučelju)*: Pogledajte što se dogodi kada postavimo pokazivač u bilo koju ćeliju tablice. Kada postavimo pokazivač u bilo koju ćeliju tablice, pojavit će se kontekstualna kartica Alati za tablice (Table tools) koja u sebi ima još dvije kartice: Dizajn (Design) i Raspored (Layout). One će nam trebati malo kasnije.

# Označavanje ćelije, redaka i stupaca te cijele tablice [7min]

N: Sad ćemo krenuti na označavanje redaka, stupaca, ćelija te cijele tablice. To će nam isto trebati kasnije kod umetanja i brisanja redaka, stupaca i ćelija te promjene visine i širine redaka i stupaca. Okrenite se svi da vidite kako se to radi.

#### Označavanje stupaca

N *(demonstracija – Slika 1: Demonstrirat ću na primjeru tablice koju sam kreirala.)*: Kao i pri uređivanju teksta dio tablice koji želimo uređivati ili oblikovati moramo najprije označiti. Za označavanje stupca postavljamo pokazivač miša iznad stupca koji želimo označiti i, kada se pokazivač okrene i zacrni, kliknemo. Označeni stupac bit će jasno istaknut.

+				
		0		

#### Slika 4 Označavanje stupaca

#### Označavanje redaka

N *(demonstracija – Slika 2: Demonstrirat ću na primjeru tablice koju sam kreirala.):* Ako želimo označiti cijeli redak, postavljamo pokazivač miša lijevo od retka koji želimo označiti. Miš će se okrenuti udesno. Kliknemo i redak će biti označen.

+				
8				
5				
$\sim$				
			90	

#### Slika 5 Označavanje redaka

#### Označavanje ćelije

N *(demonstracija – Slika 3: Demonstrirat ću na primjeru tablice koju sam kreirala.):* Ako pak želimo označiti ćeliju, postavljamo pokazivač miša uz lijevi rub ćelije. Kliknemo kada se pokazivač okrene udesno i kada se zacrni. Označena ćelija bit će jasno istaknuta.

*			

Slika 6 Označavanje ćelije

Označavanje više stupaca, redaka ili ćelija

N (demonstracija – Slika 4, Slika 5, Slika 6: Demonstrirat ću na primjeru tablice koju sam

*kreirala.*): Ponekad trebamo označiti više redaka, stupaca ili ćelija. Nakon što smo označili pojedinačnu ćeliju, redak ili stupac, ne otpuštajući lijevu tipku miša, pomičemo pokazivač preko odabranog područja. Kada smo označili područje koje želimo, otpuštamo lijevu tipku miša.

#### Slika 7 Označavanje više stupaca

÷			
		2 3	2
1			

Slika 8 Označavanje više redaka

11			
			6
8			
3		5 	V.
9			
- 14		-	

#### Slika 9 Označavanje više ćelija

N: Dakle, za označavanje stupca se postavljamo iznad stupca koji želimo postaviti dok nam se strelica ne okrene prema dolje i ne zacrni. Za označavanje retka se postavljamo sa lijeve strane retka koji želimo označiti dok nam se strelica ne okrene zrcalno. Za označavanje ćelije se postavljamo u donji lijevi kut ćelije dok se strelica ne okrene zrcalno i ne zacrni. Kad dobijemo željeni oblik strelice (ovisno označavamo li stupac, redak ili ćeliju), kliknemo lijevu tipku miša.

N: Ako želimo označiti više redaka, stupaca ili ćelija, držimo lijevu tipku miša i povlačimo lijevo-desno ako su u pitanju stupci, gore-dolje ako su u pitanju retci, dok za ćelije možemo ići i

lijevo-desno i gore-dolje. Lijevu tipku miša puštamo onda kada smo označili sve što smo htjeli označiti.

N *(Demonstrirat ću na primjeru tablice koju sam kreirala.)*: Koristeću koju tipku na tipkovnici možemo odabrati više redaka, stupaca ili ćelija koji su međusobno odvojeni?

# Označavanje cijele tablice

N *(demonstracija – Slika 7: Demonstrirat ću na primjeru tablice koju sam kreirala.):* Krenimo na označavanje cijele tablice. Označavanje cijele tablice možemo napraviti klikom na četverosmjernu strelicu iznad lijevog ruba tablice.

++				
	~			
1				
				7

# Slika 10 Označavanje cijele tablice

# Postavljanje širine i visine stupaca i redaka i promjena veličine cijele tablice [13min]

**N:** Sad ćemo krenuti na postavljanje širine i visine stupaca i redaka i promjenu veličine cijele tablice. Nakon stvaranja tablice svi stupci imaju istu širinu, a svi retci istu visinu. Visinu i širinu redaka i stupaca u tablici možemo mijenjati i određivati na nekoliko načina.

# Postavljanje širine stupaca i visine redaka pomoću miša

**N** *(Demonstrirat ću na primjeru prve tablice koju sam kreirala.)*: Poslušajte i pogledajte koji je prvi način. Prvi način je postavljanje širine stupaca i visine redaka uz pomoć miša.

- Za početak se uključi ravnalo odabere se kartica Prikaz (View) i označi se opcija Ravnalo (Ruler).
- Pokazivač miša se postavi na rub stupca ili retka kojima se želi promijeniti širinu ili visinu. Miš će poprimiti posebni oblik.
- Držeći pritisnutu lijevu tipku miša i prateći promjenu u tablici i na ravnalu može se promijeniti veličina stupca i retka prema potrebi.

N: Vaš sljedeći zadatak je da prvoj tablici koju ste prethodno kreirali, koristeći opisani način promijenite veličinu parnih redaka i neparnih stupaca na novu veličinu po vlastitom izboru. [3min]

# Tablica 5 Rješenje zadatka 3

**N:** Ima li netko tko nije uspio promijeniti visinu i širinu redaka i stupaca? Ako ste svi završili, onda možemo krenuti na još jedan način postavljanja širine stupaca i visine redaka.

Postavljanje širine stupaca i visine redaka iz grupe naredaba Veličina ćelija (Cell Size)

**N** (*Demonstrirat ću na primjeru druge tablice koju sam kreirala.*): Poslušajte i pogledajte korake drugog načina. Širinu stupaca i visinu redaka možemo postaviti i u grupi naredaba

Veličina ćelija (Cell Size).

- Označi se jedan ili više stupaca ili redaka čiju se veličinu želi mijenjati.
- Prikaže se karticu Raspored (Layout) i grupu naredaba Veličina ćelije (Cell Size).
- Klikom na strelice naredaba Visina (Height) (redak) ili Širina (Width) (stupac) promijeni se veličina redaka i stupaca.

**N:** Sljedeći zadatak glasi ovako. U drugoj tablici koju ste kreirali, koristeći opisani način promijenite veličinu parnih redaka na 3cm i neparnih stupaca na 1cm. **[4min]** 

# Tablica 6 Rješenje zadatka 4

N: Ima li netko tko nije uspio promijeniti širinu stupaca i visinu redaka na ovaj način?

N: Koju razliku uočavate između prvog i drugog načina postavljanja širine stupaca?

# Promjena veličine cijele tablice

N (*Demonstrirat ću na primjeru prve tablice koju sam kreirala.*): Poslušajte i pogledajte koji je postupak promjene veličine cijele tablice.

- Postavi se pokazivač miša na tablicu i pričeka se dok se ne pojavi mali kvadratić u donjem desnom kutu.
- Postavi se pokazivač miša na njega i miš će poprimiti oblik dvostrane strelice.
- Klikne se na kvadratić i miš će poprimiti oblik crnog križića. Ne otpuštajući lijevu tipku miša, pomiče se pokazivač dok se ne postigne željena veličina tablice.

N: Vaš sljedeći zadatak je da koristeći opisani način promijenite veličinu prve tablice. [2min]

# Tablica 7 Rješenje zadatka 5

N: Ima li netko tko nije uspio promijeniti veličinu tablice na ovaj način?

(Provjeravam učenike jesu li završili zadatke 3, 4, i 5.)

# Umetanje redaka i stupaca [8min]

N (Demonstrirat ću tako što ću kreirati tablicu koju će učenici kreirati kasnije i u nju ću dodati stupce i retke jednako kako će oni kasnije trebati u zadatku.): Poslušajte i pogledajte način na koji se mogu umetati novi retci i stupci. Često je u tablicu potrebno umetnuti dodatne retke i stupce. Word to omogućuje na jednostavan način.

- Označi se stupac ili redak pokraj kojeg se želi umetnuti novi.
- Iz kartice Raspored (Layout) u grupi naredaba Reci i stupci (Rows and Columns) izabere se što se želi umetnuti i način umetanja.

N: Stupce možete umetati lijevo ili desno od odabranog stupca, a retke iznad ili ispod od odabranog retka.

**N:** Ovdje je potrebno napomenuti da ako označimo više stupaca ili redaka, bit će umetnuto onoliko novih stupaca ili redaka koliko smo prethodno označili. Na primjer, ako označimo 3 retka i izaberemo umetanje novih, bit će umetnuta 3 nova retka.

N: Vaš sljedeći zadatak je: [5min]

 a) Kreirajte tablicu na jedan od dva načina koja smo naučili ispod tablica koje već postoje u vašem dokumentu. Tablica neka ima 4 stupca i 6 redaka.

Tablica 8 Rješenje zadatka 6a

 b) Postavite se u drugi stupac i dodajte 2 stupca – jedan lijevo, jedan desno od odabranog stupca.

# Tablica 9 Rješenje zadaka 6b

c) Postavite se u drugi redak i dodajte 2 retka – jedan iznad, jedan ispod odabranog retka.

# Tablica 10 Rješenje zadatka 6c

N: Ima li netko tko nije uspio dodati retke i stupce? Ako ste svi gotovi, onda ćemo krenuti dalje.

# Umetanje ćelija [5min]

# N (Demonstrirat ću u novoj (trećoj) tablici koju sam kreirala ono što oni u zadatku trebaju

*napraviti.*): Sad ću pokazati kako se mogu umetati ćelije, pa pogledajte. Osim redaka i stupaca, možemo umetati i ćelije. Za umetanje ćelija ćemo kliknuti na ikonu pokretača dijaloškog okvira grupe Reci i stupci (Rows and Columns). Otvorit će se dijaloški okvir Umetanje ćelija (Insert Cells) u kojemu možemo izabrati način umetanja.

**N:** U tablicu koju ste kreirali u prethodnom zadatku, umetnite dvije ćelije koristeći opciju Pomakni ćelije desno. **[3min]** 

Tablica 11 Rješenje zadatka 7

N: Što se dogodilo sa tablicom nakon umetanja novih ćelija?

N: Ima li netko tko nije uspio dodati ćelije?

(Provjeravam učenike jesu li završili zadatke 6 i 7.)

# Brisanje stupaca i redaka [6min]

N (Demonstriram na prvoj tablici koju sam kreirala ono što će učenici kasnije riješiti u

*zadatku.*): Sad kad smo naučili umetanje stupaca i redaka, naučit ćemo i brisanje tih stupaca i redaka. Poslušajte i pogledajte kako se brišu stupci i retci. Postoji nekoliko načina brisanja, ali mi ćemo ovdje opisati postupak brisanja stupaca ili redaka koristeći grupu naredaba Reci i stupci (Rows and Columns).

- Označi se stupac ili redak koji se želi izbrisati.
- Iz kartice Raspored (Layout) u grupi naredaba Reci i stupci (Rows and Columns) klikne se na gumb Izbriši (Delete).
- Iz padajućeg izbornika se izabere naredba Izbriši stupce (Delete Columns) ili Izbriši retke (Delete Rows).

N: Vaš sljedeći zadatak je da u prvoj tablici obrišete: [3min]

a) 2 stupca po izboru

# Tablica 12 Rješenje zadatka 8a

b) 2 retka po izboru

Tablica 13 Rješenje zadatka 8b

# N: Što se dogodilo sa tablicom nakon brisanja stupaca?

**N:** Ima li netko tko nije uspio obrisati retke i stupce? Ako ste svi uspjeli, onda nastavljamo dalje sa brisanjem ćelija i brisanjem cijele tablice. Pogledajte načine na koje to možete napraviti.

# Brisanje ćelija [2min]

N *(Demonstriram na drugoj tablici u svojem dokumentu ono što će učenici kasnije trebati napraviti u zadatku.)*: Osim redaka i stupaca, možemo brisati i ćelije. Ako na gumbu Izbriši (Delete) izaberemo mogućnost Brisanje ćelija (Delete Cells), otvorit će se dijaloški okvir Brisanje ćelija (Delete Cells) koji nudi nekoliko različitih načina brisanja.

# Brisanje cijele tablice [5min]

N (Demonstriram na trećoj tablici koju sam kreirala u svojem dokumentu, ali je ne brišem do kraja nego ostavljam otvoreni okvir nakon desnog klika na kvadratić i pokazivač na opciji za brisanje cijele tablice.): Ako pak želimo obrisati cijelu tablicu, mišem se postavimo iznad lijevog ruba tablice i na četverosmjernu strelicu kliknemo desnom tipkom miša. Odaberemo opciju Obriši tablicu (Delete table).

N: Vaš sljedeći zadatak je da u drugoj tablici obrišite jednu ćeliju koristeći opciju Pomakni ćelije lijevo (Shift cells left). Nakon toga obrišite svoju treću tablicu. [3min]



Tablica 14 Rješenje zadatka 9 (brisanje ćelija)

N: Što se dogodilo sa tablicom nakon brisanja ćelija?

N: Ima li netko tko nije uspio obrisati ćelije i tablicu?

# Završni dio

U završnom dijelu sata ću ponoviti s učenicima što smo sve naučili. Neka pitanja će učenici odgovarati tako što će dolaziti za glavno računalo i demonstrirati. Kada ponovimo što smo radili (pritom ću pratiti odgovore učenika te svaki točan odgovor pohvaliti i ponoviti), do kraja sata će dobiti samostalan zadatak izrade rasporeda sati koji će nastaviti na sljedećem satu ako ne stignu do kraja danas. Tokom ponavljanja ću koristiti dokument

METOD2\_Matekovic\_Ana\_JP1\_pitanja\_odgovori.doc u kojem se nalaze sva pitanja i odgovori. Dok će rješavati svoj zadatak, na platnu će biti prikazano gotovo rješenje (isti primjer koji sam pokazala na početku sata dok sam objašnjavala zaglavlja redaka i stupaca sa treće stranice iz dokumenta METOD1\_Matekovic\_Ana\_JP1\_primjeri.doc).

N: Pri kraju smo sa satom pa ćemo ponoviti što smo danas sve naučili.

# Ponavljanje [15min]

Zvjezdica prije pitanja znači da je za to pitanje moguća demonstracija učenika za glavnim računalom (prethodno ću otvoriti novi dokument za ponavljanje).

N: Što je tablica?							
N: Koji su osnovni dijelovi tablice?							
N: Što je redak?							
N: Što je stupac?							
N: Što je ćelija?							
*N: Kako umećemo tablicu?	*N: Koji je drugi način?						
*N: Kako označavamo cijeli stupac?							
*N: Kako označavamo cijeli redak?							
*N: Kako označavamo ćeliju?							
*N: Kako označavamo više stupaca, redaka ili ćelija?							
*N: Kako označavamo cijelu tablicu?							
*N: Kako postavljamo širinu stupaca i visinu	*N: Koji je drugi način?						
redaka?							
*N: Kako mijenjamo veličinu cijele tablice?							
*N: Kako umećemo nove retke i stupce?	N: Gdje možemo umetnutni nove retke?						

N: Gdje možemo umetnuti nove stupce?

\*N: Kako umećemo nove ćelije?

\*N: Kako brišemo stupce, retke ili ćelije?

\*N: Kako brišemo cijelu tablicu?

# Samostalni zadatak

Minimalno pet minuta prije kraja zadajem samostalni zadatak.

N: Sad ćete krenuti sa samostalnim zadatkom. Radite ćete do kraja sata, a što ne stignete danas, završit ćete sljedeći put.

N: Vaš zadatak je napraviti svoj raspored sati. Za početak razmislite koliko vam je potrebno stupaca i redaka, a kad počnete raditi, koristite elemente koje smo danas naučili. Također možete koristiti druge oblike slova, veličine, podebljanja, pozadinske boje i slično. Sjetite se samo što smo ponovili na početku sata, što smo radili danas i iskoristite maštu.

N: Koliko redaka, a koliko stupaca će vaša tablica na kraju imati?

**N:** Krenite sad raditi, pa koliko stignete danas. Ako imate kakvih pitanja, slobodno pitajte. Prije zvona spremite svoj zadatak na mjesto gdje i inače spremate, a nazovite ga *Prezime\_Ime\_RS*.

(Jednu minutu do zvona ću dati uputu za kraj sata.)

N: Uskoro će zvoniti, pa vas molim da prije izlaska iz razreda isključite računala i lijepo spremite svoje radno mjesto. Molim vas nemojte ostaviti smeće na stolu i oko stola nego lijepo sve počistite. Hvala.

N: Doviđenja i ugodan dan vam želim!

U: Doviđenja!

Kraj pripreme

#### 3.2. Primjer pisane pripreme 2. – javno predavanje

Sadržaj ovog nastavnog sata, odrađen je na **Šestoj osnovnoj školi Varaždin** u **sedmom razredu.** Tema nastavnog sata su tablične kalkulacije u **MS Excellu**. Za primijetiti je kao i kod prethodne pripreme da se sadržaj pripreme svodi na vrlo detaljne upute i više zajedničkog rada nego samostalnog rada.

Početak pripreme:

Studijski profil: Informatika u obrazovanju

Škola – vježbaonica: VI OŠ Varaždin

Datum: 28.11.2014.

Razredni odjel: 7

Redni broj nastavnog sata:

Nastavna cjelina /tema: MS Excel

Nastavna jedinica: Formati brojeva u ćelijama tablice

#### 1. Cilj nastavnog sata:

Definirati i prepoznati format brojeva u ćelijama tablice

#### 2. Zadaće nastavnog sata:

a) Obrazovne zadaće

Učenici će:

- prepoznati različite formate brojeva
- definirati formate brojeva u ćeliji
- izraditi tablicu sa različitim formatima ćelija

#### b) Funkcionalne zadaće

#### Unaprijediti vještine:

- rukovanja računalom pri izradi proračunskih tablica
- razvijati sposobnost samostalnog oblikovanja broja u ćeliji tablice
- preciznog izvođenja naredbi oblikovanja brojeva u ćeliji tablice
- razvijati sposobnost povezivanja, zaključivanja i uspoređivanja naučenog korištenja Excel-a

#### c) Odgojne zadaće

Poticati učenike:

- samostalnost pri radu u Excelu
- razvijanje osjećaja osobne vrijednosti pravilnim odabirom naredbi
- međusobnu suradnju i pomaganje
- poštivanje utvrđenih pravila komunikacije

#### 3. Tip nastavnog sata

Obrada novog sadržaja

#### 4. Nastavni oblici

Frontalna nastava, individualni rad

#### 5. Nastavne metode

Metoda razgovora, metoda demonstracije, metoda praktičnih radova

#### 6. Nastavna sredstva

računalo, LCD projektor, zbirka zadataka

# 7. Izvori znanja

- Babić M., Dimovski Z., Glavan F., Leko S., Stančić M., Vejnović B. (2014.) «Moj portal», Grafički zavod Hrvatske

## Organizacija nastavnog rada – artikulacija metodičke jedinice

Redni	Faze rada i sadržaj	Metodičko oblikovanje	Vrijeme u minutama
broj			
1.	Uvodni dio	Razgovor	10
	Predstavljanje	Frontalni rad	
	Ponavljanje	Individualni rad	
2.	Središnji dio	Frontalni rad	77
	Obrada novog sadržaja:	Demonstracija	
	-Formati brojeva u ćelijama	Razgovor	
	Demonstracija primjera	Individualni rad	
	Vođena vježba		
3.	Završni dio	Razgovor	
	- Ponavljanje	Individualni rad	3

#### UVODNI DIO SATA (10 min)

Pozdravim učenike i predstavim se.

- N: Dobar dan!
- U: Dobar dan!

N: Ja sam X.X. i održat ću vam ovaj sat, nadam se da ćemo dobro surađivati. Molim vas ako imate kakvih pitanja tijekom sata ili znate odgovor na postavljeno pitanje da dignete ruku.

N: Danas ćemo raditi u Excel-u, koje dokumente možemo izraditi u Excelu.

U: Cjenike, račune, razne tablice.

N: Zašto za izradu tih dokumenata koristimo Excel a ne Word, u kojem isto možemo praviti tablice.

U: Podatke u Excelu možemo zbrajati, oduzimati, napraviti grafikone ..

N: Sad ćemo ponoviti što ste do sada naučili analizirajući prikazanu tablicu .

N: Otvaram (METOD2\_Došlić\_Zvjezdana\_JP2, Radni list Troškovi) i zumiranjem prikazujem tablicu Troškovi.

N: Koliko stupaca ima prikazana tablica?

U: Dva.

- N: Koliko redova ima prikazana tablica?
- U: Deset.
- N: Koja je adresa označene ćelije?
- U: B3.
- N: Koji oblik imaju podatci upisani u tablicu?
- U: Brojevi i tekst
- N: Po kojem rubu su poravnati brojevi?
- U: Desno
- N: Po kojem rubu je poravnat tekst?
- U: Lijevo

N: Uključite računala i pokrenite program Excel. Tko nije uspio pokrenuti Excel?

N: Na svom računalu samostalno izradite prikazanu tablicu. Obilazim učenike, odgovaram na postavljena pitanja, provjeravam tko nije završio zadatak. (vrijeme: 5 min)

N: Tablicu ćemo spremiti u vašu mapu (ako je imaju) ili radnu površinu pod nazivom Troškovi.xls

#### SREDIŠNJI DIO SATA (75 min)

Najava teme: Danas ćemo učiti formate brojeva u ćelijama tablice.

Nastavnik piše na ploču: Formati brojeva u ćelijama tablice.

N: Prepoznajete li što je ovo? (Učenicima pokazujem cjenik.)

U: Cjenik

N: Da li ovakav cjenik možemo izraditi u Excelu?

U: Možemo.

N: Ako neki proizvod košta 5 kuna i 25 lipa na koji način se piše cijena tog proizvoda?

U: 5,25 kn.

N: Kada provjeravamo rok trajanja proizvoda, u kojem obliku nam piše datum.

U: 25.3.2002.

Na isti način možemo oblikovati i brojeve koje smo unijeli u našu tablicu.

Molim vas sad samo pratite nemojte ništa raditi u svom radnom listu, na primjeru pokazat ću oblikovanje brojeva u ćeliji. Demonstriram učenicima primjer na nastavničkom računalu. (METOD2\_Došlić\_Zvjezdana\_JP2, Radni list Troškovi) Vrijeme: 4 min

Postupak:

- Označimo ćeliju koju želimo oblikovati C6
- Desna tipka miša dijaloški okvir oblikovanje ćelija odaberemo karticu broj
- Ponuđene su nam kategorije po kojima možemo oblikovati ćelije
- Odaberemo po kojoj kategoriji želimo oblikovati Kako se u ovoj tablici radi o novčanim vrijednostima odabrati ćemo kategoriju Valutno
- Simbol ćemo odabrati kuna, dva decimalna mjesta
- Izbor potvrđujemo odabirom tipke OK

N: Isto oblikovanje možemo napraviti na još dva načina pomoću kartice broj. Demonstriram učenicima još 2 načina. Možete koristiti način koji vam se čini najlakši.

N: Sad ćemo na vašu tablicu primijeniti ova oblikovanja, vraćam sa na tablicu Troškovi i vodim učenike korak po korak kroz valutno oblikovanje ćelija. (Vrijeme: 3 min)

N: Tko nije uspio izvršiti zadatak. Pomažem učenicima koji nisu uspjeli.

N: Ako u ćeliju upišemo bez oblikovanja tekst i brojeve, da li u njoj možemo izvršavati računske operacije?

U: Ne.

N: Kako Excel definira tu ćeliju?

U: Kao tekst.

N: U vašoj tablici Troškovi u ćeliji C12 zbrojite ukupan iznos i oblikujte ga valutno, valuta kuna, bez decimalnih mjesta.

Obilazim učenike, odgovaram na postavljena pitanja, provjeravam tko nije završio zadatak.

N: U ćeliji C14 izračunajte koliko je ostalo neplaćeno. Kako ćemo to izračunati?

U: Od ćelije C12 oduzet ćemo ćeliju C13.

Obilazim učenike, odgovaram na postavljena pitanja, provjeravam tko nije završio zadatak.

N: Dobiveni rezultat ćemo oblikovati kao broj sa dva decimalna mjesta, koji se ako je negativan prikazuje crveno. Taj način prikaza koristimo ako želimo prikazati neki dug, da nas crvenom bojom upozori.

Molim vas sad samo pratite nemojte ništa raditi u svom radnom listu, na primjeru pokazat ću oblikovanje broja u ćeliji. Demonstriram učenicima na nastavničkom računalu i verbalno pratim. (vrijeme 3 min)

N: Upisani broj oblikovati ćemo na sljedeći način:

Postupak:

- Označimo ćeliju koju želimo oblikovati C14
- Desna tipka miša dijaloški okvir oblikovanje ćelija odaberemo karticu broj
- Odaberemo po kojoj kategoriji želimo oblikovati –odabrati ćemo kategoriju Brojčano
- Odaberemo vrstu negativni brojevi i razdjelnik tisućica.
- Izbor potvrđujemo odabirom tipke OK

N: Samostalno svojoj ćeliji C14 promijenite format u brojčani, negativne brojeve prikažite crveno, sa dva decimalna mjesta.

Provjeravam tko nije završio zadatak, prozivam jednog učenika da demonstrira i verbalno prati na nastavničkom računalu na koji je način riješio zadatak. (vrijeme 3 min)

N: Tablici Troškovi dodajte još jedan stupac kako je prikazano (METOD2\_Došlić\_Zvjezdana\_JP2, radni list Datum) i upišite datume plaćanja. Dane, mjesece i godine razdvajate točkom, pazite da bi upisane datume mogli oblikovati nakon godine ne smijete unijeti točku. Ako unesemo točku nakon godine Excel podatak neće prepoznati kao datum.

(vrijeme: 4 min)

N: Tko nije uspio izvršiti zadatak. Pomažem učenicima koji nisu uspjeli.

Molim vas sad samo pratite nemojte ništa raditi u svom radnom listu, na primjeru pokazat ću oblikovanje datuma u ćeliji. (METOD2\_Došlić\_Zvjezdana\_JP2, radni list Datum). Demonstriram učenicima na nastavničkom računalu i verbalno pratim. (vrijeme 3 min)

N: Upisane datume plaćanja oblikovati ćemo na sljedeći način:

Postupak:

- Označimo ćeliju koju želimo oblikovati D6
- Desna tipka miša dijaloški okvir oblikovanje ćelija odaberemo karticu broj
- Odaberemo po kojoj kategoriji želimo oblikovati –odabrati ćemo kategoriju Datum
- Odaberemo vrstu datuma i jezik
- Izbor potvrđujemo odabirom tipke OK

N: Sad ćemo na vašu tablicu primijeniti ova oblikovanja, vraćam sa na tablicu (METOD2\_Došlić\_Zvjezdana\_JP2, radni list Datum) i vodim učenike korak po korak kroz datumsko oblikovanje ćelija.

N: U tablici Troškovi oblikujte datume u ćelijama od D7-D11 svaki drugom vrstom datumskog oblikovanja.

Obilazim učenike, odgovaram na postavljena pitanja, provjeravam tko nije završio zadatak. (vrijeme 3 min)

N: Sada će zvoniti, rad ćemo nastaviti sljedeći sat.

N: Podatke, koje smo u ćeliji oblikovali kao datume možemo jednostavno oduzimati i zbrajati. Na taj način izračunati ćemo koliko ima dana do Nove godine. Demonstriram učenicima izračun na nastavničkom računalu. (METOD2\_Došlić\_Zvjezdana\_JP2, radni list Datum, vrijeme: 3 min)

Dobiveni rezultat oblikovat ću kao brojčanu vrijednost, bez decimalnih mjesta.

N: Na isti način izračunajte koliko dana ste stari. Izračun napravite na istom radnom listu na kojem se nalazi tablica Troškovi. Ispod tablice Troškovi kreiraj tablicu: Koliko si star.( METOD2\_Došlić\_Zvjezdana\_JP2, radni list Datum). Dobiveni rezultat oblikujte brojčano bez decimalnih mjesta. (Vrijeme izrade je 5 min.)

Obilazim učenike, odgovaram na postavljena pitanja, provjeravam tko nije završio zadatak.

N: Tablici Troškovi dodajte još jedan stupac kako je prikazano i upišite vrijeme plaćanja, kad unosimo vrijeme u ćeliju potrebno je sate i minute razdvojiti dvotočkom. (METOD2\_Došlić\_Zvjezdana\_JP2, radni list: Vrijeme)

Molim vas sad samo pratite nemojte ništa raditi u svom radnom listu, na primjeru pokazat ću oblikovanje vremena u ćeliji. Demonstriram na nastavničkom računalu. (vrijeme: 3 min)

N: Upisano vrijeme oblikujemo na sljedeći način:

Postupak:

- Označimo ćeliju koju želimo oblikovati E6
- Desna tipka miša dijaloški okvir oblikovanje ćelija odaberemo karticu broj
- Odaberemo po kojoj kategoriji želimo oblikovati –odabrati ćemo kategoriju Vrijeme
- Odaberemo vrstu prikaza i jezik
- Izbor potvrđujemo odabirom tipke OK

N: U svoju ćeliju E6 upišite vrijeme 7:45, sad ćemo zajedno oblikovati vrijeme, vodim učenike korak po korak kroz demonstraciju.

Provjeravam tko nije upio riješiti zadatak.

N: Probajte što će se dogoditi ako ćeliju C6 oblikujemo kao tekst.

U: Izgubila je oznaku valute.

N: Po kojem rubu se poravnala?

U: Po lijevom.

N: Excel tekst poravnava po lijevom rubu, a brojeve po desnom.

N: Na isti način kako smo u prethodnim zadatcima oblikovali ćelije brojčano, valutno, datumski možemo ih oblikovati i postotno. U ćeliju G7 upišite broj 50, upisani broj oblikujte postotno istim postupkom kao u prethodnim zadatcima koristeći kategoriju postotak. (vrijeme: 2 min) Obilazim učenike, po potrebi im pomažem u radu. Provjeravam tko nije završio zadatak.

N: Tko će pokazati na koji je način je riješio zadatak? Učenik koji se javi demonstrira i verbalno prati oblikovanje ćelije u obliku postotka.

N: Spremite svoju radnu knjigu naredbom Spremi kao u mapu sa svojim imenom pod nazivom oblikovanja.xls.

Učenicima dajem im upute za rad: Potrebno je samostalno izraditi prikazanu tablicu (METOD2\_Došlić\_Zvjezdana\_JP2, radni list: Vježba), u sve ćelije unijeti broj 4,5 i izvršiti tražena oblikovanja. Vrijeme za izradu zadatka je 7 min učenici samostalno rješavaju zadatak, obilazim učenike i po potrebi im pomažem u radu.

Provjeravam tko nije završio zadatak?

Prozivam jednog učenika da na nastavničkom računalu demonstrira rješenje prvog oblikovanja, drugi učenik demonstrira rješenje drugog oblikovanja...., rješenja verbalno pratim, učenici prate i kontroliraju svoj rad, po potrebi ga korigiraju.

N: Zadatak ćemo spremiti u vašu mapu (ako je imaju) ili radnu površinu pod nazivom Oblikovanje vježba.xls
# ZAVRŠNI DIO SATA (3 min)

Dajem učenicima uputu da isključe računala, i postavljam nekoliko pitanja za ponavljanje.

- N: Na koji način excel poravnava brojeve , a na koji tekst?
- U: Brojeve desno, a tekst lijevo.
- N: Koje vrste oblikovanja brojeva u ćeliji smo danas naučili?
- U: Naučili smo brojčano oblikovanje, datumsko, valutno i oblikovanje vremena.
- N: Kako Excel razdvaja decimalne brojeve, a kako tisućice?
- U: Decimalne brojeve zarezom, a tisućice točkom.
- N: Kako razdvajamo podatke u ispisu datuma?
- U: Točkom.
- N: Kako razdvajamo podatke u ispisu vremena?
- U: Dvotočkom
- N: Zahvaljujem se na suradnji, uskoro će zvoniti, spremite svoja radna mjesta. Doviđenja.

# Kraj pripreme

# **3.3.** Primjer pisane pripreme 3. – javno predavanje

Idući nastavni sat, održan je na **Drugoj osnovnoj školi Varaždin**. Tema nastavnog sata je **mreža i internet**. Sama tema nije dosta praktična da bi se mogla primijeniti u obliku konkretnih zadataka kao kod primjera oblikovanja teksta ili tabličnih kalkulacija. Samo kreativnost nastavnika ovaj nastavni sat koji se više temelji na teorijskim postavkama, je učinila sat zanimljivim i motivacijskim za učenike. Na nastavnom satu, nastavnik je osigurao aktivno sudjelovanje učenika u usvajanju teme mreže i internet uključujjući učenike u rasapravu i pretraživanje pojmovlja interneta i mreže.

## Početak pripreme:

Škola: II. Osnovna škola, Varaždin

Razred: 6. razred

Nadnevak izvođenja nastavnog sata: 15.01.2015.

Nastavna cjelina: Mreže i internet

Nastavna tema: Povezivanje računala

Tip sata: obrada novog gradiva

Cilj nastavnog sata:

Upoznati računalne mreže te razumijeti princip mrežne komunikacije.

#### Materijalni zadaci:

- Razumijeti pojam računalne mreže
- Razlikovati podjele računalnih mreža
- Razumijeti način komunikacije računala
- Utvrditi značaj uloga računala u mreži
- Upoznati prednosti i nedostatke primjene računalnih mreža

# Funkcionalni zadaci:

- Poticati interes za samostalno otkrivanje ostalih mogućnosti računalnih mreža
- Osposobiti za samostalno korištenje mrežnih mogućnosti

#### Odgojni zadaci:

- Stjecati pozitivan stav prema radu
- Razvijati suradnju i međusobno pomaganje
- Osvijestiti o važnosti primjene računalnih mreža

#### Nastavna sredstva i pomagala:

- Računalo
- Projektor

#### Nastavne metode:

- Metoda razgovora
- Metoda usmenog izlaganja
- Metoda praktičnog rada

# Oblici rada:

- Frontalni rad
- Samostalni rad
- •

# Literatura:

MOJ PORTAL 3.0, 7 udžbenik informatike s višemedijskim nastavnim materijalima u šestom razredu osnovne škole, Autori: Magdalena Babić, Zoran Dimovski, Fredi Glavan, Mario Stančić, Branko Vejnović; Nakladnik: ŠK-poglavlja 4.1. "Povezivanje računala"

REDNI	·	METODIČKO	VRIJEME U
BROJ	FAZE RADA I SADRZAJ	OBLIKOVANJE	MINUTAMA
1. 2.	UVODNI DIO - Pozdravljanje i predstavljanje - Dogovor o načinu rada	<ul> <li>Usmeno izlaganje</li> <li>Razgovor</li> </ul>	5'
	SREDIŠNJI DIO -Definiranje računalnih mreža -Prikaz mrežnih mogućnosti -Prikaz prednosti i nedostataka mrežnog rada -Definiranje, navođenje i prikaz zlonamjernih programa -Obrazloženje računalne komunikacije -Definiranje i demonstriranje uloga računala u mreži -Obrazloženje i prikaz vrsta računalnih mreža -Definiranje i prikaz medija za prijenos podataka	<ul> <li>Usmeno izlaganje</li> <li>Razgovor</li> <li>Demonstracija</li> <li>Individualni rad</li> </ul>	37'

3.	ZAVRŠNI DIO		
	<ul><li>Ponavljanje</li><li>Pozdravljanje</li></ul>	- Razgovor	3'

# 1. UVODNI DIO (5 minuta)

Predstavljam se i dogovaram pravila rada te dajem uputu da isključe sve nepotrebne sadržaje (youtube, igrice i ostale internetske izvore) te da isključe računalne monitore.

N:Dobar dan.

U:Dobar dan.

N:Moje ime je Zoran Hercigonja i danas ću biti vaš nastavnik.

N:Ako imate bilo kakvih pitanja ili želite nešto reći, podignite ruku i pitajte.

N: Jeste li čuli za pojam: računalna mreža?

U: Jesmo.

N: Koja je najpoznatija računalna mreža na svijetu?

U: Internet.

N: Tako je. Internet ili drugim nazivom World Wide Web je najpoznatija računalna mreža koja nam pruža niz mrežnih mogućnosti.

N: Što sve možete raditi na internetu?

U: Pristupati Facebooku i računalnim igricama.

N: Da. Računalne mreže omogućavaju komunikaciju putem socijalnih mreža kao što je na primjer Facebook, zatim pristupanje različitim sadržajima u međuostalom pristupanju igricama, ali omogućavaju i prijenos podataka putem elektroničke pošte.

N: Zamislimo da se u jednom trenutku prilikom naših aktivnosti na internetu prekine veza.

N: Htjeli smo se naći s prijateljima na Facebooku malo prokomentirati sadržaje ili odigrati umreženu igru i najedanput nam je ta opcija postala nedostupna.

N:Prekid veze može nastati bilo gdje na mreži.

N: Prekid može nastati zbog problema u našem računalu, a može biti problem u udaljenom računalu.

N: Nije uvijek običaj da se nedostupnost mreže javlja samo zato što nemamo signala ili zato što mrežni kabel nije dovoljno dobro priključen u mrežni utor.

N: Da bismo mogli utvrditi mjesto prekida mreže, moramo se upoznati s načinom mrežne komunikacije te vrstama mreža koje osiguravaju tu komunikaciju i medijima prijenosa podataka.

N: Kako bi identificirati sve moguće vrste prekida na mrežama, moramo prvo definirati sam pojam računalne mreže.

## 2. SREDIŠNJI DIO SATA(trajanje: 37minuta)

#### (Otvaram prezentaciju Mreže.pttx)

N: Da bi stigli u školu krećemo se cestom ili ulicom koja povezuje školu s mjestom stanovanja.

N: To znači da su škola i mjesto stanovanja povezani putem kojim svakodnevno putujemo u školu i vraćamo se kući. Na taj način se ostvaruje povezivanje škole i mjesta stanovanja.

N: Ako posjećujemo rodbinu ili prijatelje na različitim mjestima, putujemo putovima koji povezuju naše mjesto stanovanja s njihovim mjestima stanovanja. Na taj način ostvarujemo mrežu poveznica.

N:Spomenuli smo Facebook preko kojeg je moguća komunikacija s drugim korisnicima.

N: Da li moramo biti povezani s tim korisnicima da bi mogli ostvariti komunikaciju?

U: Moramo biti povezani.

N: Tako je. S korisnicima moramo biti povezani kako bi ostvarili komunikaciju.

N: Ta poveznica je zapravo put kojim naši podaci putuju do drugog korisnika i obrnuto.

N: Kako smo nazvali grupu poveznica kada smo govorili o posjećivanju rodbine i prijatelja?

U: Nazvali smo je mrežom.

N: Odlično. Nazvali smo je mrežom.

N: Komunikaciju preko Facebooka ostvariti ćemo preko računala. Komunicirajući s više prijatelja, stvaramo mrežu komunikacija.

N: Kako bismo onda na temelju toga definirali što je to računalna mreža?

U: Računalna mreža znači da su računala povezana.

N: Tako je. Računalna mreža znači da su računala međusobno povezana.

#### (Referenciram se na Mreže.pttx, slajd 2)

N: No treba naglasiti da se radi o više računala pa računalnu mrežu definiramo kao skup povezanih računala koja mogu međusobno komunicirati, ali i razmjenjivati podatke.

N: Koliko je najmanje računala potrebno da bi se stvorila računalna mreža?

U: Potreba su dva računala.

N: Tako je. Da bi se ostvarila komunikacija i razmjena podataka između računala, potrebno je uključiti u mrežu najmanje dva računala.

N: U informatici se računalne mreže još definiraju kao sustav strojne i programske opreme.

N: Što sve koristimo u radu s računalom?

U: Koristimo monitor, miš, tipkovnicu, zvučnike.

N: Bravo. Uz računalo koristimo monitor, miš, tipkovnicu i zvučnike koji predstavljaju strojnu opremu računala.

N: Uz to dolaze još i kabeli i žice koje omogućavaju računalu da dobiva električnu struju kako bi mogao raditi.

N: Koje alate na računalu najčešće koristite?

U: Koristimo: Word, Excel, Paint.

N: Dobro. Word, Excel i Paint su aplikacije koje je netko od infromatičara isprogramirao pa ih nazivamo programskom opremom računala.

N: Koje aplikacije koristimo za komunikaciju?

U: Facebook i elektroničku poštu.

N: Tako je. Facebook i elektronička pošta su neke od aplikacija za ostvarivanje komunikacije.

N: Pripadaju li i one u programsku opremu računala?

U: Pripadaju.

N: Tako je. Pripadaju jer su i one isto programirane aplikacije koje je izradio informatičar.

N: Programsku i strojnu opremu koristimo da bi ostvarili komunikaciju između računala.

N:Pogledajmo na sljedećoj slici sve potrebne elemente mrežne komunikacije. (Demonstriram primjer na **Mreže.pttx, slajd 3**).

N: Za komunikaciju su prije svega potrebna dva računala koja istovremeno mogu biti i primatelji i pošiljatelji.

N: Na primjer kada komuniciramo s drugom osobom, istovremeno smo i slušač, ali i govornik. Prvo slušamo što druga osoba govori, a onda odgovaramo ili pitamo.

N:Isto je i s računalima. I ona su istovremeno slušači i govornici.

N: Osim toga potreban je i medij za prijenos podataka.

N: Recimo najpoznatiji medij za prijenos podataka je papir. S papira možemo čitati napisane riječi, a možemo napisati i nove riječi i rečenice.

N: Vaše radne bilježnice iz informatike, isto su mediji za prijenos podataka.

N: Što sadrže radne bilježnice?

U:Radne bilježnice sadrže zadatke.

N: Tako je. Radne bilježnice sadrže zadatke, ali i mjesta predviđena za zapisivanje novih informacija u obliku rješenja zadataka.

N: Mediji za prijenos podataka u računalnoj mreži mogu biti žični i bežični. No o njima ćemo detaljnije govoriti malo kasnije.

N: Sljedeći element računalne komunikacije je protokol.

N: U svakodnevnom životu se susrećemo s pravilima ponašanja.

N: Što su pravila ponašanja?

U: Ona nam govore kako se trebamo ponašati.

N: Tako je. Pravila nas nam govore kako bi se trebali ponašati.

N: I u informatici postoje pravila ponašanja mrežne komunikacije.

N: Njih nazivamo protokolom.

N: U svakodnevnom životu postoji bonton ili skup pravila ponašanja koja propisuju kako se treba ponašati.

N: Koje pravilo primjenjujemo kad sretnemo nastavnika u hodniku ili učioni?

U: Pozdravljamo nastavnika.

N: Tako je. Bonton nam nalaže da je pristojno pozdraviti nastavnika kad uđe u učionicu ili kada ga sretnemo na hodniku.

N: Protokol je isto neka vrsta informatičkog bontona koji predstavlja skup pravila za prijenos podataka i informacija.

N: Koje pravilo vrijedi kada jedna osoba govori?

U: Druga je osoba sluša i ne prekida je.

N: Tako je. Kada jedna osoba govori, druga je sluša bez prekidanja i obrnuto.

N: U informatici umrežena računala komuniciraju na način da pošiljatelj pošalje zahtjev primatelju, a primatelj na taj zahtjev odgovara pošiljatelju.

N: To je jedno od pravila u protokolu za prijenos podataka. Uvijek mora postojati zahtjev na temelju kojeg primateljsko računalo može odgovoriti pošiljatelju. Naravno protokoli podrazumijevaju i propisanu količinu podataka koju je moguće slati preko računalne mreže. N: I kao posljednji element mrežne komunikacije je komunikacijski kanal koji predstavlja dvosmjernu komunikaciju između računala primatelja i pošiljatelja.

N: Putem tog kanala se provodi komunikacija i razmjena podataka.

N: Što smo rekli za što se koriste Facebook i elektronička pošta?

U: Koriste se za komunikaciju.

N: Dobro. Koriste se za komunikaciju. No što još možemo slati putem elektorničke pošte osim poruke?

U: Možemo slati datoteku.

N: Tako je. Možemo slati i datoteku.

# (Referenciram se na Mreže.pttx, slajd 4)

N: Mrežne mogućnosti su stoga komunikacija i slanje podataka najčešće u obliku datoteka.

N: To jednom rječju nazivamo resursima.

N: Razmjena resursa podrazumijeva slanje i primanje podataka, ali sto tako i zajedničko korištenje pisača na primjer.

N: To znači da više osoba čija su računala umrežena, imaju zajednički pristup pisaču. Na primjer da u ovoj učionici postoji jedan pisač umrežen s računalima, svi bismo mogli ispisati željene sadržaje, jer svi imamo prava pristupa pisaču.

N: Računalna mreža nam pruža i smještanje podataka u slučaju kvara. Uvijek je dobro imati kopiju zbog sigurnosnih razloga.

N: Da li možemo pristupiti podacima ako računalo prestane raditi?

U: Ne možemo pristupiti podacima.

N: Da. Kada računalo ne radi, ne možemo pristupiti podacima. Zato je dobro da se podaci nalaze na mjestu kojem je moguće pristupiti bilo kad u bilo kojem trenutku.

N: Poznata aplikacija koja nudi mogućnosti spremanja podataka je Google Drive (Google disk). (*Referenciram se na Mreže.pttx, slajd 5, 6.*)

N: Putem računalne mreže možemo smjestiti podatke na jedno mjesto i tako se osigurati u slučaju da se dogodi kvar na našem računalu koji bi uzrokovao gubitak svih važnih podataka. N: Google Drive pruža mogućnost spremanja, ali i preuzimanja podataka.

N: Osim toga tu su već spomenute komunikacijske mogućnosti koje osiguravaju komunikaciju korisnika koji su fizički udaljeni jedni od drugih.

N: Neke poznate aplikacije su Facebook, Skype, Twitter. (Referenciram se na Mreže.pttx, slajd 7)

N: Dakle prednosti korištenja mrežnog rada su iznimno velike. Svi sadržaji su nam dostupni u u bilo koje vrijeme s bilo kojeg mjesta u obliku vijesti, vremenske prognoze, on-line kupnje.

N: Osim računala s kojim se još uređajima možemo spojiti na internet?

U: možemo se spojiti preko mobitela i tableta.

N: Tako je. Poznati pametni telefoni i tableti pružaju mogućnost spajanja na internet.

N: Računalne mreže nam pružaju i mobilni internet. (Referenciram se na Mreže.pttx, slajd 8)

N: Ovo su bile prednosti.

N: Koja je najpoznatija opasnost koja vreba na internetu?

U: Najpoznatija opasnost su virusi.

N: Da. Računalni virusi ili zlonamjerni programi vrlo često smanjuju sigurnost podataka, ali i sigurnost mrežne komunikacije.

N: Eto uspjeli smo identificirati prvi slučaj zbog kojeg računalna mreža ne radi. Veza je prekinuta zbog toga što je računalo zaraženo virusom ili nekim drugim oblikom zlonamjernog programa. Dakle prekid je nastao zbog kvara na računalu zaraženom virusima.

N: Osim virusa postoje Spyware i Spam. (Referenciram se na Mreže.pttx, slajd 9)

# Zadatak 1 (trajanje 2 minute):

N: Uključite monitore i koristeći se računalnom mrežom (Internetom), potražite značenje pojmova Spam i Spyware.

N: Pokrenite jedan od preglednika (Internet Explorer, Mozilla ili Google Chrome) koji se nalaze na radnoj površini računala i u pretraživač upišite pojmove Spyware i Spam.

- 155 -

( Dok pretražuju, prođem razredom i provjeravam njihov rad. Nakon isteka 2 minute prozovem nekoliko učenika da pročitaju definicije Spywarea i Spama)

N: Dobro. Spyware je široka kategorija softwarea s namjenom da presreće ili preuzima djelomično kontrolu rada na računalu bez znanja ili dozvole korisnika. *(Referenciram se na Mreže.pttx, slajd 10)* 

N: Vidimo sada kako komunikacija računala može biti narušena presretačima koji mogu preuzeti djelomičnu kontrolu nad računalom bez znanja korisnika. Nismo ni svijesni da internet ne radi zato što neki zlonamjerni program ne dopušta komunikaciju s udaljenim računalom.

N: Gdje se pojavljuje Spam kao napad na sigurnost mrežne komunikacije?

U: Spam se pojavljuje kod elektroničke pošte.

N:Tako je. Spam kao opasnost vreba na elektroničkoj pošti. Dakle Spam je neželjena elektronička pošta koja može u velikom opsegu zatrpati primateljev elektronički sandučić za primanje poruka. Naravno Spam može i blokirati promet elektroničke pošte pa opet nismo u stanju razmjeniti podatke.(*Referenciram se na Mreže.pttx, slajd 11*)

# Zadatak 2 (trajanje 2 minute):

N: Spomenuli smo maloprije računalne viruse. Na internetu pronađite definiciju računalnih virusa i najpoznatije računalne viruse.

N: Pokrenite jedan od preglednika (Internet Explorer, Mozilla ili Google Chrome) koji se nalaze na radnoj površini računala i u pretraživač upišite pojam Računalni virus.

( Dok pretražuju, prođem razredom i provjeravam njihov rad. Nakon isteka 2 minute prozovem nekoliko učenika da pročitaju definiciju računalnog virusa te da nabroje neke poznate viruse)

N:Računalni virusi su programi koji mogu "zaraziti" druge programe tako da kopiraju samoga sebe unutar drugih programa. Neki najpoznatiji virusi su Trojanski konj, Worms, Code Red, World Concept. (*Referenciram se na Mreže.pttx, slajd 12*).

N: Dakle pri nekom prethodnom pristupanju internetu, računalo se zarazilo virusom, a da nismo ni primjetili. Trenutačni anitvirusni programi nisu uopće ni identificirali taj virus i on je počeo kopirati samoga sebe unutar drugih programa i tako nam onemogućio spajanje na internet. (Dajem uputu da isključe ponovno monitore)

N: Na principu kojeg brojevnog sustava rade računala?

U: Na principu binarnog brojevnog sustava.

N: Tako je. Računala rade na principu binarnog brojevnog sustava.

N: Koje brojeve rabe pritom?

U: Rabe nulu i jedinicu.

N: Tako je. Računala rabe sustav nula i jedinica za prijenos podataka.

N: Računalna komunikacija se izvodi preko zapisa nula i jedinica koji se zapisuju u spremnike ili registre.

N: Pogledajmo sljedeću sliku. Sadržaji u obliku 0 i 1 prenose se iz spremnika jednog računala preko prijenosnog kanala (u ovom slučaju prijenosni kanal je označen strelicom) u spremnik drugog računala.

N: Ako računala imaju spremnike za prijenos podataka, da li medij za prijenos ima isto spremnik?

U: Medij za prijenos ima isto spremnik.

N: Tako je. Medij za prijenos također sadržava spremnik za pohranu sadržaja. (*Referenciram se na Mreže.pttx, slajd 13 i 14*)

N: Sadržaj se može izgubiti na putu između spremnika računala i spremnika medija za prijenos i time onemogućiti komunikaciju računala te razmjenu podataka između korisnika.

N: Sadržaj nije stigao do računala kojem je bio namijenjen i komunikacija nije ostvarena.

N: Kod pretraživanja pojmova Spywer, Spam i računalni virus, koju ulogu je imalo vaše računalo? Da li je imalo ulogu poslužitelja ili korisnika?

U: Računalo je imalo ulogu korisnika.

N: Tako je. Pretraživanjem sadržaja vi ste bili korisnik koji je šaljući zahtjev tražio od poslužitelja isporučivanje podataka za zadani pojam. Komunikacija između računala podrazumijeva upravo dvije uloge: korisničku i poslužiteljsku ulogu.

N: Koju biste ulogu imali kada bi netko od vas tražio da mu pošaljete elektorničkom poštom domaću zadaću?

U: Imali bi ulogu poslužitelja.

N: Tako je. Imali bi ulogu poslužitelja.

N: Upravo zbog tih uloga razlikujem odnose računala u mreži kao nervanopravan (tzv. client-server) i ravnopravan (tzv. peer-to-peer) odnos. (*Referenciram se na Mreže.pttx, slajd 15*).

N: Zamislimo da jedno računalo sadržava programe i podatke, a drugo računalo samo pristupa tim programima i podacima.

N: Da li bi računala bila u ravnopravnom ili neravnopravnom odnosu?

U: Računala bi bila u neravnopravnom odnosu.

N: Tako je. Računala bi bila neravnopravna. Zbog čega?

U: Jedno računalo ima sve programe i podatke, a drugo nema.

N: Bravo. Vrlo dobar odgovor. Jedno računalo sadržava podatke i programe, a drugo nema ništa osim mogućnosti pristupanja tim programima i podacima.

N: Kako smo jednom rječju nazvali sve podatke koje računalo sadržava?

U: Nazvali smo ih resursima.

N: Tako je.Sve podatke koje sadrži jedno računalo, nazvali smo resursima.

N: U neravnopravnom odnosu svako računalo ima svoju ulogu što znači da korisnička računala ne mogu posluživati zato što nemaju resurse kojima bi posluživali, dok je poslužiteljsko računalo to koje samo poslužuje određenim resursima korisnička računala.

N: Primjerice glavno računalo sadržava sve pohranjene podatke i programe koje kao uslugu pruža na korištenje korisnikovu računalu svaki puta kada to korisnik zatraži *(Referenciram se na Mreže.pttx, slajd 16)* 

N: S obzirom na zadatke poslužitelja, koji bi onda bili zadaci korisničkog računala?

U: Preuzimanje podataka.

N: Tako je preuzimanje, ali i spremanje podataka na poslužitelj.

N: Kada bi primjerice na spomenutu aplikaciju Google Drive (Google disk) postavili neke datoteke, onda bi to značilo da smo na poslužitelj predali ili pospremili podatke.

N: Pogledajmo na prezentaciju. U sredini se nalazi poslužiteljsko računalo koje je ovim dvostrukim strelicama povezano s korisničkim računalima. Dvostruke strelice označavaju preuzimanje i pospremanje podataka.

N: Što bi označavala strelica označena isprekidanom linijom?

U: Strelica označena isprekidanom linijom predstavlja komunikaciju.

N: Tako je. Strelica označena isprekidanom linijom označava komunikaciju dva korisnička računala preko glavnog poslužiteljskog računala. (*Referenciram se na Mreže.pttx, slajd 17 i 18*)

N: Da li bi bila moguća komunikacija i slanje podataka kada poslužitelj ne bi radio?

U: Komunikacija i slanje podataka ne bi bila moguća.

N: Tako je. Ako bi se poslužiteljsko računalo pokvarilo, komunikacija korisnika ne bi bila moguća, a time i preuzimanje resursa to jest podataka i programa.

N: Kod neravnopravnog odnosa računala napomenuli smo da jedno računalo sadrži resurse, dok drugo može samo pristupati tim resursima.

N: Kako bi onda izgledao ravnopravan odnos računala?

U: Sva računala bi imala resurse.

N: Tako je. Dakle tu ne postoji računalo koje bi sadržavalo sve programe i podatke u obliku resursa za posluživanje, već svako računalo u mreži ima resurse kojima komunicira.

N: Ravnopravni odnos nazivamo i peer-to-peer što u prijevodu znači točka prema točki ili računalo prema računalu.

N: Dakle sva računala su povezana po ravnopravnoj osnovi, ravnopravno komuniciraju jedno s drugim.

N: Ako su računala u ravnopravnom odnosu, koje uloge onda ima pojedino računalo?

U: Svako računalo ima ulogu korisnika i poslužitelja.

N: Tako je. Svako računalo je ujedno u ulozi korisnika i poslužitelja. Tu nema opasnosti od prestanka rada glavnog poslužiteljskog računala jer ono kao takvo ne postoji. *(Referenciram se na Mreže.pttx, slajd* **19**).

N: Naravno u ovakvoj situaciji gdje su sva računala u ulozi poslužitelja i korisnika, može doći do pretopterećenja računalne veze i s time do prekida rada.

N: Dakle, poznata je situacija kada istovremeno računala međusobno razmjenjuju podatke da dođe do preopterećenja veze i time do prekida. Kanali prijenosa su toliko opterećeni da sav sadržaj nije u mogućnosti biti poslan korisniku.

N: U razredu svako računalo ima svoje resurse.

N: Koje uloge posjeduju računala?

U: Računala posjeduju ulogu poslužitelja i korisnika.

N: Tako je. Svako računalo u razredu ima svoje resurse: programe i podatke.

N: U kakvom su odnosu računala ovdje unutar razreda?

U: U ravnopravnom odnosu.

N: Tako je. U razredu su računala u ravnopravnom odnosu.

N: Identificirali smo još jedan slučaj prekida veze.

N: Prekid veze može se dogoditi unutar iste zgrade ili institucije, što nazivamo lokalni prekid veze ili izvan jedne zgrade ili institucije što nazivamo globalni prekid veze.

N: Prema tome osnovni kriterij za podjelu računalnih mreža je prema veličini odnosno prema lokaciji umreženih računala. (*Referenciram se na Mreže.pttx, slajd 20*)

N: Stoga razlikujemo LAN, MAN i WAN mreže.

N: Pogledajmo ovu sliku (Referenciram se na Mreže.pttx, slajd 21)

N: Dakle lokalna mreža se može nalaziti na razini škole ili stambene zgrade. Više stambenih zgrada ili škola je povezano na veću mrežu imenom MAN mreža koja obuhvaća područje grada ili više gradova.

N: Računalne mreže na razini država povezane u jednu svjetsku mrežu predstavljaju WAN računalnu mrežu.

N: LAN računalna mreža ili engleski Local Area Network je lokalna mreža do nekoliko desetaka računala u istoj zgradi ili prostoriji.

N: LAN sadrži tri topologije odnosno tri načina povezivanja računala. To su: BUS ili sabirnička topologija, TOKEN RING ili prstenasta i STAR ili zvjezdasta topologija. (*Referenciram se na Mreže.pttx, slajd 22*).

N:Dakle u sabirničkoj topologiji, sva računala su spojena na jednu sabirnicu ili vod.

N: Posebnost ove topologije je u tome što samo računala kojima su upućeni podaci prihvaćaju te podatke, dok ostala računala odbacuju podatke koji im nisu upućeni.

N: Pogledajmo primjer na prezentaciji. (Referenciram se na Mreže.pttx, slajd 23).

N: Računalo 2 prije slanja "sluša" da li netko od ostalih računala šalje podatke.

N: Ako nitko ne šalje podatke, Računalo 2 započinje s procesom slanja ili SENDING.

N: Računalo 5 prepoznaje da je poruka za njega i prihvaća je odnosno izvodi se proces RECEIVE.

N: Budući da druga računala nisu slala podatke za vrijeme procesa nije se dogodilo ometanje odnosno COLLISION i proces je uspješno završen.

N: Krajevi sabirnice imaju takozvane završetke koji prekidaju signale. To je važno zbog toga što se signali mogu vratiti te dovesti do ometanja ili COLLISION. To bi dodatno opterećivalo mrežu.

N: Što bi se dogodilo sa signalom kada ga niti jedno od računala ne bi prihvatilo?

U: Signal bi bio prekinut.

N: Bravo. Signal bi bio prekinut.

N: Gdje bi se dogodilo prekidanje signala?

U: Prekidanje signala bi se dogodilo na završecima.

N: Tako je. Signal bi bio prekinut na završecima.

N: Sljedeća topologija je TOKEN RING ili prstenasta topologija.

N: TOKEN RING ili prstenasta topologija podrazumijeva da su računala povezana po principu da svako računalo ima prethodika i sljedbenika.

N: Podaci se šalju prvom računalu do sebe, ali samo u jednom smjeru. Podatke će prihvatiti računalo čija je adresa navedena uz podatke. (*Referenciram se na Mreže.pttx, slajd 24*).

N: Zamislimo da sada prosljeđujete udžbenik susjedu do sebe i tako u krug.

N: Što će se dogoditi ako se netko od vas izmakne?

U: Udžbenik će pasti na pod.

N: Tako je. Udžbenik će pasti na pod i prijenos će se završiti.

N: Dakle prekinuti je prsten i udžbenik neće stići do učenika kojem je i namijenjen.

N: Što će se dogoditi ukoliko jedno računalo prestane raditi?

U: Prekinuti će se komunikacija.

N: Tako je. Budući da se radi o jednosmjernoj komunikaciji i pritom jedno računalo prestane raditi, prekida se prsten i podaci nikad neće dospjeti do korisnika kojem su bili namijenjeni.

N: Događa se isto kao i kod primjera s prosljeđivanjem udžbenika.

Ovo je prikaz prekida veze koji se može javiti na lokalnoj razini odnosno na razini jedne zgrade ili prostorije.

N: Sljedeća topologija je STAR topologija ili zvjezdasta topologija koja podrazumijeva da su sva računala povezana na jedan zajednički medij kojeg nazivamo zvjezdište. Dakle komunikacija i prijenos podataka se odvija isključivo preko zvjezdišta. (*Referenciram se na Mreže.pttx, slajd 25*).

N: Takvu smo komunikaciju spominjali kod neravnopravnog odnosa računala.

N: Koju ulogu bi imalo računalo u zvjezdištu?

U: Računalo u zvjezdištu bi imalo ulogu poslužitelja.

N: Bravo. Računalo bi imalo ulogu poslužitelja, jer preko njega komuniciraju sva preostala računala.

N: Što će biti s vezom ako bi se računalo u zvjezdištu pokvarilo?

U: Računalna veza bi bila prekinuta.

N: Da. Računalna mreža bi bila svakako prekinuta, jer ne bi postojalo središte preko kojeg se vrši komunikacija.

N: Da li se u razredu nad računalima primjenjuju pojedinačne topologije ili kombinacija svih triju?

U: Upotrebljava se kombinacija.

N: Tako je. Kombinacijom triju topologija smanjuje se postotak prekida računalne veze.

N: Ukoliko se dogodi prekid na jednoj od topologija, ostale topologije će osigurati nesmetanu komunkaciju i razmjenu podataka.

N:Nadalje MAN mreža ili engl. Metropolitan Area Network podrazumijeva rasprostranjenu računalnu mrežu na području jednog grada ili više gradova. Dakle ona objedinjuje sve LAN računalne mreže.

N: Primjerice sve lokalne LAN mreže u školama na području grada Varaždina se objedinjuju u MAN mrežu. (*Referenciram se na Mreže.pttx, slajd 26).* 

N: Svaku LAN mrežu sada možemo promatrati kao jednu točku ili jedinicu.

N: Isto kao što su u LAN mreži računala predstavljala pristupne točke odnosno jedinice, tako u MAN mreži manje lokalne mreže predstavljaju pristupne jedinice ili točke.

N: Ukoliko povežemo nekoliko MAN računalnih mreža u veću mrežu, dobiti ćemo WAN mrežu engl. Wide Area Network ili mrežu na širem geografskom području. *(Referenciram se na Mreže.pttx, slajd 27).* 

N: Sada su više MAN mreža pristupne točke ili jedinice koje tvore WAN mrežu.

N: WAN mreža podrazumijeva povezivanje cijelog svijeta.

N: U koju računalnu mrežu pripada onda Internet?

U: Internet pripada u WAN mrežu.

N: Tako je Internet pripada u WAN mrežu.

N: Zbog čega?

U: Jer povezuje cijeli svijet.

N: Tako je. Internet povezuje sve MAN i LAN mreže u jednu zajedničku.

N: Istražili smo moguće prekide veza na razini zlonamjernih programa, topologija, odnosa i uloga računala. Sada vidimo koliko je zapravo složena situacija kada dođe do kvara na mreži i prekida komunikacije.

N: Kvar ne mora uvijek nužno značiti da je uzrok prekida smješten lokalno unutar iste zgrade ili da je uzrok prekida smješten u našem računalu. Prekid se može dogoditi i na širem području računalnih mreža u obliku prekida MAN ili WAN mreža. N: Prekid rada ovisi i o mediju za prijenos podataka putem računalne mreže.

N: Kako su u učionici povezana računala s računalnom mrežom?

U: Računala su povezana pomoću kabela.

N: Tako je. Računala u učionici su povezana pomoću mrežnih kabela.

N: Ovdje je medij prijenosa mrežni kabel odnosno žična veza.

N: Koji još mediji za prijenos podataka postoje?

U: Postoji i bežični medij.

N: Tako je. Bežični mediji su izvedeni u obliku Routera ili mrežnog adaptera.

N: Pogledajmo sad kako izgledaju mediji za prijenos podataka.

N: Prva vrsta medija je žična veza.

N: Računala su fizički povezana s mrežnim kabelima. *(Referenciram se na Mreže.pttx, slajd 28)*. Mrežni kabeli najčešće imaju osam vodova za prijenos podataka.

N: Mrežni kabel se sastoji od UTP konektora i koaksijalnih vodova s PVC izolacijom. (*Referenciram se na Mreže.pttx, slajd 29, 30*).

N: Koaksijalni vod se sastoji od vanjske zaštitne izolacije, bakrenog opleta, unutarnje izolacije i pobakrene čelične žice. (*Referenciram se na Mreže.pttx, slajd 31*)

N: Prednost takvog medija je čvrsta i postojana veza kod koje rijetko može doći do gubitka signala.

N: Bi li došlo do prekida veze ukoliko je mrežni kabel fizički oštećen?

U: Došlo bi do prekida veze.

N: Tako je. Veza bi bila prekinuta ako bi mrežni kabel imao jaka fizička oštećenja koja bi onemogućavala prolazak signala kroz njega.

N: Oštećenje se može dogoditi na UTP konektoru ili na jednom od bakrenih vodova.

N: Kao medij prijenosa podataka i komunikacije, postoje i bežične mreže koje nastaju povezivanjem računala bez fizičkih veza. One su izvedene u obliku mrežnih adaptera, to jest uređaja koji preusmjeravaju mrežne signale.

N: Mrežni adapter odašilje određene signale.

N: Bi li veza bila prekinuta ukoliko bi došlo do miješanja signala?

U: Veza bi bila prekinuta.

N: Tako je. Takve veze izvedene preko bežićne mreže vrlo često mogu biti ometane drugim signalima i zbog ometanja može doći do gubitka podataka ili prekida. (*Referenciram se na Mreže.pttx, slajd 32*).

N:Među novijim medijima za prijenos nalazi se optički kabel ili svjetlovod koji se sastoji od optičkih vlakana kroz koje podaci putuju u obliku svjetlosti.

N: Svjetlovod je jako sličan koaksijalnom vodu.

N: I svjetlovod ima zaštitni omotač ili izolaciju, zatim plašt jezgre koji predstavlja unutarnju izolaciju i jezgru od sitnih optičkih vlakana kroz koje putuje svjetlost. *(Referenciram se na Mreže.pttx, slajd 33, 34)*.

N: Ovakav medij osigurava siguran i iznimno brz prijenos podataka.

N: Znamo da je brzina svjetlosti najveća poznata brzina. Stoga podaci prenošeni tom brzinom predastavljaju iznimno brz prijenos.

N: Budući da su svjetlovodi vrlo slični koaksijalnom vodu, uslijed čega bi se dogodio prekid veze?

U: Prekid veze bi se dogodio uslijed oštećenja svjetlovoda.

N: Tako je. Ako bismo fizički oštetili svjetlovod, veza bi bila prekinuta jer bi oštećenja onemogućila prolazak svijetla kroz svjetlovod.

N: Danas smo jako puno toga rekli o računalnim mrežama i mrežnoj komunikaciji. Pa ponovimo što smo sve naučili.

## 3. ZAVRŠNI DIO (3 minute)

N: Koliko je računala potrebno za mrežnu komunikaciju?

U: Potrebna su najmanje dva računala.

N: Tako je. Za mrežnu komunikaciju su potrebna najmanje dva računala.

N: U kakvim odnosima mogu biti računala u mreži?

U: Računala mogu biti u ravnopravnim i neravnopravnim odnosima u mreži.

- 165 -

N: Tako je računala mogu biti u ravnopravnim i neravnopravnim odnosima.

N:Nabrojite vrste računalnih mreža prema veličini!

U: Mreže prema veličini su LAN, MAN, WAN.

N: Točno. Mreže prema veličini su LAN, MAN, WAN.

N: Koje smo mrežne topologije spominjali uz LAN mrežu?

U: Spominjali smo BUS, TOKEN RING i STAR topologiju.

N: Točno. Spominjali smo BUS ili sabirničku topologiju, TOKEN RING ili prstenastu topologiju i STAR ili zvjezdastu topologiju.

N: Hvala na suradnji. Bili ste dobri. Možete na odmor.

N: Doviđenja.

U: Doviđenja.

Zvono

Kraj pripreme

# 3.4. Primjer pisane pripreme 4. – javno predavanje

Sadržaj pripreme, namijenjen je izvođenju nastavnog sata učenicima sedmog razreda Šeste osnovne škole Varaždin. Tema je praktičnog karaktera, a tiče se izrade grafikona u aplikaciji za tablične kalkulacije MS Excell.

Početak pripreme:

Škola: VI. osnovna škola Varaždin Razred: 7.d Nastavni predmet: Informatika Nastavna cjelina: Proračunske tablice - Microsoft Excel 2013 Nastavna jedinica: Grafikoni Tip sata: obrada novog gradiva Blok sat: da Datum: 9. prosinac 2014.

## PLAN VOĐENJA (ORGANIZACIJE) NASTAVNOG PROCESA

## Cilj obrade nastavne jedinice

Upoznavanje učenika s primjenom različitih oblika grafikona koristeći alat MS Excel 2013.

Zadatci koje treba ostvariti da bi se postigao cilj

Materijalni zadatci:

- razlikovati skupove podataka u tablici
- razlikovati kada koristiti koji tip grafikona
- odrediti važnost podataka u skupu podataka

• argumentirati primjenu određenog tipa grafikona za zadani skup podataka Funkcionalni zadatci:

- stvoriti grafikon, premjestiti ga u novi list i promijeniti veličinu
- primijeniti različite stilove oblikovanje grafikona
- dodati naslov, oznake osi, legendu i pozadinu grafikona

Odgojni zadatci:

- formirati pozitivne stavove prema radu
- razvijati smisao za urednost, upornost i ustrajnost u radu
- poticati međusobno pomaganje i suradnju
- poticati samostalnost prilikom demonstracija

#### Nastavne metode

- 4. Metoda razgovora
- 5. Metoda usmenog izlaganja
- 6. Metoda praktičnih aktivnosti

#### Oblici rada

- 7. Frontalni rad
- 8. Individualni rad

## Nastavna sredstva i pomagala

- 9. Računalo
- 10. Projektor

# 11. MS Excel 2013

# ORGANIZACIJA NASTAVNOG RADA – ARTIKULACIJA METODIČKE JEDINICE (makroplan)

REDNI	FAZE RADA I SADRŽAJ	METODIČKO	VRIJEME U
BROJ		OBLIKOVANJE	MINUTAMA
1.	UVODNI DIO	16.Usmeno izlaganje	10'
	12 Pozdravljanje i upoznavanje s učenicima	17 Frontalni rad	
	(kratko predstavljanje da ću ja danas održati	17.11011011111100	
	nredavanje)		
	predavanjej		
	13. Dogovor o načinu rada		
	14 Dependionic line used is now predice		
	14. Ponavljanje kao uvod u novo gradivo		
	15. Predstavljanje sadržaja nastavne jedinice		
2.	SREDIŠNJI DIO	19. Usmeno izlaganje	35'
2.1.	Demonstracije i zadatci	20. Demonstracija	
	18. Nastavnikova demonstracija pojedine	21. Praktična	
	funkcionalnosti i kratki primjeri za	aktivnost	
	uvježbavanje (samostalni rad učenika, po	22 Individualni rad	
	potrebi učenikova demonstracija rješenja		
	na nastavnikovom računalu)	23. Frontalni rad	
	VELIKI ODMOR	24.15	
2.2.	Zadatak za vježbu	28. Praktična aktivnost	40'
	25. Obrada zadataka koji objedinjuju sve	29.Individualni rad	
	naučeno na prethodnom satu	30 Demonstracija	
	(temperaturne anomalije, dobit, proizvodi)	30.Demonstracija	
	26. Samostalno riešavanie zadatka od strane		
	učenika po navedenim koracima riešavanja		
	···· · [- · · · · · · · · · · · · · · ·		
	27. Učenička demonstracija rješenja nakon		
	svakog koraka		
3	ZAVRŠNI DIO	35 Frontalni rad	5'
5.			5
	31. Ponavljanje		
	SZ. INAJAVA ZVUIIA		

	<ol> <li>Napomena učenicima da pospreme radna mjesta</li> </ol>	
	34. Pozdrav	

TIJEK IZVOĐENJA NASTAVE – NASTAVNI RAD (mikroplan)

# 1. Uvodni dio (10 minuta)

U ovom dijelu sata se predstavljam učenicima, dogovaram način rada, radim ponavljanje kao uvod u novo gradivo i učenicima predstavljam što će se raditi.

# N: Dobar dan!

# U: Dobar dan!

N: Ja sam Zoran Lotina; student Fakulteta organizacije i informatike i danas ću vam održati dva sata nastave iz informatike. Danas ću ja biti vaš učitelj.

Učenike upoznajem s pravilima rada.

*N:* Pravila vrijede i dalje ista, tako da ako imate pitanje ili odgovor - podignite ruku.

Najavljujem današnju temu grafikoni. Prikazujem sadržaj nastavne jedinice (datoteka grafikoni.xlsx, stranica Sadržaj).

Prije obrade novog gradiva postavljanjem pitanja s učenicima ponavljam što su do sada radili.

N: Prije nego što počnemo s obradom novog gradiva, ponovimo što ste do sada naučili u Excelu.

Ako nakon 4 sekunde nema podignute ruke dajem jedan primjer (u upitnom obliku) što su učenici radili kako bih ih aktivirao.

# N: *Što je tablica?*

Organizirani način prikaza podataka složen u retke i stupce. (Rešetka)

N: Što je redak?

Učenike podsjećam na ono što su radili na matematici. Koordinatni sustav u ravnini. Apscisa – os x (vodoravni pravac).

# N: Što je stupac?

Učenike podsjećam na ono što su radili na matematici. Koordinatni sustav u ravnini. Ordinata – os y (okomiti pravac).

N: Što je ćelija?

Presjek stupca i retka.

Učenike podsjećam na ono što su radili na matematici. Koordinatni sustav u ravnini. Točka – uređeni par (x,y).

N: Što možemo upisati u ćelije?

Broj, tekst, formulu.

N: Koje format brojeva ste do sada upoznali?

Datum, vrijeme, postotak.

Učenike motiviram za rad pokazujući im konačan izgled tablica *Temperaturne\_anomalije*, *Dobit* i *Proizvodi* time istodobno pokazujući koji će biti rezultat njihovog rada. Da se samome sadržaju da smisao kroz praktičan primjer s kojim se učenici susreću u svakodnevnom životu, a sada će to i izraditi.

N: Ovi grafikoni sadržavaju većinu elemenata grafikona navedenih u sadržaju nastavne jedinice.

Ako vam se ovo na prvi pogled čini nerazumljivo, ne brinite se; prvo ćemo zajedno proći svaki od elemenata grafikona, dobit ćete i zadatke za isprobavanje svakog od elementa prikaza grafikona, a slijedeći sat ćete na temelju naučenoga izraditi ove grafikone na temelju pripadajućih tablica.

Napominjem kakav je način rada.

N: Za svaki element grafikona naveden u sadržaju (dajem prikaz sadržaja na projekciji - datoteka grafikoni.xlsx, stranica Sadržaj) ću vam prikazati nekoliko mogućih načina izrade. Tu demonstraciju trebate samo pratiti, tj. gledati, slušati i ako nešto nije jasno pitati. Nakon svakog od poglavlja dobit ćete zadatke na kojima ćete to isprobati, a ja ću za to vrijeme raditi provjeru rješenja.

Ponavljam napomenu o načinu rada u skraćenoj verziji da učenici znaju da demonstracije rješenja (2'-3') trebaju samo pratiti (gledati, slušati i ako nešto nije jasno pitati), a nakon toga demonstrirano primijeniti kroz zadatke, za vrijeme čega ću radi provjeru uspjeha rješavanja zadatka.

N: Ukratko; prvo ja radim vi pratite, a nakon toga vi radite ja pratim (provjeravam što radite).

Učenicima napominjem da slobodno obavezno pitaju ako im nešto nije jasno.

# 2. Središnji dio (35 + 40 minuta)

U ovom dijelu se izmjenjuju nastavnikova demonstracija svakog od elemenata grafikona navedenog u sadržaju i zadatci nakon svakog od elementa.

Uvod u demonstraciju je problemskog tipa, s pitanjima koji potiču učenike na razmišljanje, nakon čega slijede primjeri nekoliko mogućih rješenja. Učenici demonstracije rješenja (2'-3') prate, a nakon toga demonstrirano primjenjuju kroz kraće zadatke. Nastavnik radi provjeru uspjeha rješavanja zadatka i po potrebi poziva učenika koji je uspješno riješio zadatak da demonstrira rješenje ostalim učenicima na nastavnikovom računalu.

## 2.1. Vježba (35 minuta)

Učenicima predstavljam problem, demonstriram jedno rješenje, postavljam pitanje kako je to još moguće riješiti i demonstriram druge načine rješavanja.

Demonstracija i zadatci su navedeni u datoteci grafikoni.xlsx, stranica *Demos\_x*, gdje x predstavlja broj demonstracije.

U nastavku navodim načine rješavanja.

Demos 1 (3')

Umetanje grafikona

Za odabrane ćelije kartica *Umetanje*  $\rightarrow$  odjeljak *Grafikoni*  $\rightarrow$  neka od ikona grafikona (npr. preporučeni grafikoni) ili

Kartica Umetanje  $\rightarrow$  odjeljak Grafikoni  $\rightarrow$  dijaloški okvir Promjena vrste grafikona

Zadatak 1 (6')

Kreirati tablicu, upisati podatke i umetnuti grafikon.

Način rješavanja je proizvoljan.

Demos 2 (2')

Premještanje grafikona

Za odabrano područje grafikona u kontekstualnoj kartici *Alati za grafikone*  $\rightarrow$  kartica *Dizajn*  $\rightarrow$  odjeljak *Mjesto*  $\rightarrow$  ikona premjesti grafikon  $\rightarrow$  dijaloški okvir *Premjesti grafikon* ili Odabir područja grafikona desnom tipkom miša  $\rightarrow$  skočni izbornik  $\rightarrow$  element *Premjesti grafikon*  $\rightarrow$  dijaloški okvir *Premjesti grafikon* 

# Promjena veličine grafikona

Na odabrano područje grafikona postaviti pokazivač miša (kursor) na rub grafikona dok na poprimi oblik dvostrane strjelice. Držeći lijevu tipku miša pomicati pokazivač miša dok graf ne poprimi željenu veličinu (prilikom pomicanja pokazivač miša poprima oblik križića)

# Zadatak 2 (3')

Napraviti kopiju prethodno umetnutog grafikona, kopiju premjestiti u novi radni list.

Originalu promijeniti veličinu.

Način rješavanja je proizvoljan.

Demos 3 (2')

Dodavanje podataka (unutar tablice)

Dodavanje podataka (izvan tablice)

Za odabrano područje grafikona kontekstualna kartica *Alati za grafikone*  $\rightarrow$  kartica *Dizajn*  $\rightarrow$  odjeljak Podatci  $\rightarrow$  ikona *odaberi podatke*  $\rightarrow$  dijaloški okvir *Odabir izvora podataka* i odabrati podatke ili

kopirati željene ćelije i zalijepiti ih u grafikon.

Zadatak 3 (4')

Dodati novi predmet i ocjene u tablicu.

Dodati novi predmet i ocjene izvan tablice.

Način rješavanja je proizvoljan.

Demos 4 (2')

## Dodavanje naslova

Za odabrano područje grafikona u kontekstualnoj kartici *Alati za grafikone*  $\rightarrow$  kartica *Dizajn*  $\rightarrow$  odjeljak *Rasporedi grafikona*  $\rightarrow$  padajući izbornik *Dodaj element grafikona*  $\rightarrow$  odabir elementa *Naslov grafikona* ili

Odabir područja grafikona desnom tipkom miša  $\rightarrow$  skočni izbornik  $\rightarrow$  naredba *Oblikovanje područja grafikona ...* 

#### Dodavanje legende

Za odabrano područje grafikona u kontekstualnoj kartici *Alati za grafikone*  $\rightarrow$  kartica *Dizajn*  $\rightarrow$  odjeljak *Rasporedi grafikona*  $\rightarrow$  padajući izbornik *Dodaj element grafikona*  $\rightarrow$  odabir elementa *Legenda* 

Zadatak 4 (3')

Na grafikonu dodati naslov i legendu.

Demos 5 (3')

Osi

(dodavanje osi)

Za odabrano područje grafikona u kontekstualnoj kartici *Alati za grafikone*  $\rightarrow$  kartica *Dizajn* odjeljak *Rasporedi grafikona*  $\rightarrow$  padajući izbornik *Dodaj element grafikona*  $\rightarrow$  odabir elementa *Osi*.

(naslov osi)

Za odabrano područje grafikona u kontekstualnoj kartici *Alati za grafikone*  $\rightarrow$  kartica *Dizajn* odjeljak *Rasporedi grafikona*  $\rightarrow$  padajući izbornik *Dodaj element grafikona*  $\rightarrow$  odabir elementa *Nazivi osi*.

(raspon osi)

Odabir osi desnom tipkom miša (unutar područja iscrtavanja grafikona)  $\rightarrow$  skočni izbornik  $\rightarrow$  naredba Oblikovanje osi  $\rightarrow$  (sidebar) kartica *Mogućnosti osi*  $\rightarrow$  grupa *Granice* 

## Natpisi nad podatcima

Za odabrano područje grafikona u kontekstualnoj kartici *Alati za grafikone*  $\rightarrow$  kartica *Dizajn* odjeljak *Rasporedi grafikona*  $\rightarrow$  padajući izbornik *Dodaj element grafikona*  $\rightarrow$  odabir elementa *Natpisi nad podatcima*  Zadatak 5 (5')

Grafikonu dodati osi, nazive osi, podesiti raspon vertikalne osi da odgovaraju mogućim ocjenama (1-5).

Minutu prije zvona učenicima najavljujem veliki odmor. Napominjem važnost iskorištavanja odmora za osvježavanje, a ne za igranje na računalu.

Veliki odmor (15 minuta)

2. Središnji dio (40 minuta)

# 2.3. Zadatak za vježbu

Navodim učenicima što ćemo raditi i na koji način ćemo raditi. Dajem prikaz konačnog rješenja na projekciji (*Temperaturne\_anomalije, Dobit, Proizvodi*) i zatim navodim korake rješavanja. Učenicima ovo predstavljam kako igru gdje je za prelazak na slijedeći korak (razinu) potrebno izvršiti sve aktivnosti iz trenutnog koraka. Svaki zadatak se sastoji od tri koraka: izrada tablice, umetanje grafikona, uređivanje grafikona. Aktivnosti unutar koraka se razlikuju zavisno o zadatku.

N: Ono što ste naučili od prije i ono što smo zajedno obradili i što ste naučili na prethodnom satu sada ćete primijeniti na ovome zadatku. Svaki od ovih zadataka ćete rješavati kroz tri koraka: izrada tablice, umetanje grafikona, uređivanje grafikona. Svaki korak ima svoj skup aktivnosti koje je potrebno izvršiti kako bi se prešlo na slijedeći korak. Nakon rješavanja koraka, osam učenika će prezentirati svatko po jedan korak. Prilikom demonstriranja navodite i aktivnosti unutar koraka koje izvodite.

## Zadatak 1. – Temperaturne anomalije (14')



Slika 11 Prikaz konačnog rješenja zadatka Temperaturne\_anomalije

Zadatak se sastoji od izrade statističkih pokazatelja odstupanja temperature i oborina za mjesec listopad 2014. godine u obliku grafikona.

Nakon što je dio učenika riješio korak zadatka, jedan učenik demonstrira rješenje jednog koraka (korak 2. i 3.) ostalim učenicima.

#### Koraci zadatka Temperaturne\_anomalije:

Korak 1. - stvaranje tablice (5')

Popis aktivnosti:

- 36. Upisati nazive mjernih postaja
- 37. Upisati vrijednosti percentila oborina
- 38. Upisati vrijednosti percentila temperature

# Korak 2. – umetanje grafikona (1')

Popis aktivnosti

- 39. Obilježiti željene ćelije
- 40. Umetnuti prikladni tip grafikona s obzirom na skup podataka (kombinirani stupčasto-linijski)
- 41. Pozicionirati grafikon desno u odnosu na tablicu

# Korak 3. – uređivanje grafikona (8')

Popis aktivnosti

- 42. Dodati naslov grafikona
- 43. Dodati nazive osi
- 44. Podesiti raspon osi do 100
- 45. Dodati natpise nad podatcima
- 46. Podesiti razmak između stupaca grafikona



Zadatak 2. – Dobit (12')

Slika 12 Prikaz konačnog rješenja zadatka Dobit

Zadatak se sastoji od izrade prikaza prihoda, rashoda i dobiti u obliku grafikona, pri to m koristeći različite kombinacije tipova grafikona.

Nakon što je dio učenika riješio korak zadatka, jedan učenik demonstrira jedan korak rješenja ostalim učenicima.

Učenik koji demonstrira dobiva gotove podatke za prihode i rashode, ali ne formatirane.

#### Koraci zadatka Dobit:

#### Korak 1. – stvaranje tablice (6')

- 47. Unijeti mjesece prihode i rashode
- 48. Upisati formulu za dobit
- 49. Upisati formulu za ukupno

50. Formatirati prikaz brojeva u ćelijama (format valuta, negativne vrijednosti crveno)

# Korak 2. – umetanje grafikona (1')

- 51. Odabrati željene ćelije
- 52. Umetnuti prikladni tip grafikona s obzirom na skup podataka (kombinirani)
- 53. Pozicionirati grafikon desno u odnosu na tablicu

# Korak 3. – uređivanje grafikona (5')

- 54. Dodati stil grafikona (3D efekt, sjenčanje, obrub, ispuna)
- 55. Stvoriti kopiju grafikona
- 56. Kopiju grafikona premjestiti na novi radni list

# Zadatak 3. – Proizvodi (14')

					Udio c	ijene svakog proizvoda u ukupnom iznosu računa
Naziv artikla	Jmj	Kol.	Cijena	Iznos	%	4.95%
Jana	0,751	2	8,51 kn	17,02 kn	13,60%	13,60%
Zeleni čaj i limun	kom	1	6,50 kn	6,50 kn	5,19%	17,56%
Earl grey zeleni čaj	kom	1	6,50 kn	6,50 kn	5,19%	5,19%
Jucy - crvena naranča	0,33 I	3	8,99 kn	26,97 kn	21,55%	5,19%
Naturel - Mrkva i jabuka	0,51	2	19,99 kn	39,98 kn	31,95%	
Cappy - Multivitamin	0,331	2	10,99 kn	21,98 kn	17,56%	
Cappy - Jabuka nektar	0,21	1	6,19 kn	6,19 kn	4,95%	21 55%
			Ukupno:	125,14 kn	100,00%	31,95%
				Jana		🖷 Zeleni čaj i limun 🔹 Earl grey zeleni čaj 👘 Jucy - crvena naranč
				Natu	rel - Mrkva i j	abuka = Cappy - Multivitamin = Cappy - Jabuka nektar

Slika 13 Prikaz konačnog rješenja zadatka udio cijene svakog proizvoda u ukupnom račun

Zadatak se sastoji od izrade prikaza udjela cijene svakog proizvoda u ukupnom iznosu računa.

Nakon što je dio učenika riješio korak zadatka, jedan učenik demonstrira jedan korak rješenja ostalim učenicima.

Učenik koji demonstrira dobiva gotove podatke, ali ne formatirane.

# Koraci zadatka *Proizvodi*:

#### Korak 1. - stvaranje tablice (7')

- 57. Unijeti nazive proizvoda, jmj, količinu i cijenu
- 58. Upisati formulu za iznos
- 59. Upisati formulu za ukupno
- 60. Upisati formulu za izračun postotka cijene proizvoda u ukupnom iznosu računa

## Korak 2. – umetanje grafikona (1')

- 61. Odabrati željene ćelije
- 62. Umetnuti prikladni tip grafikona s obzirom na skup podataka (pita)
- 63. Pozicionirati grafikon desno u odnosu na tablicu

#### Korak 3. – uređivanje grafikona (6')

- 64. Dodati naslov grafikona
- 65. Dodati natpise nad podatcima
- 66. Dodati prozirnost odabira područja grafikona

Učenicima nisu unaprijed dane gotove tablice s podatcima, nego se od učenika očekuje napraviti tablice samostalno. Na takav način učenici istodobno ponavljaju gradivo prethodnih nastavnih jedinica.

# 7. Završni dio (5 minuta)

Ponavljanje što su učenici danas naučili. Očekivani odgovori su ekvivalent poglavlja navedenih u sadržaju nastavne jedinice. (grafikoni.xlsx, stranica *Sadržaj*)

#### N: Koje smo sve elemente grafikona u MS Excel-u naučili?

U: Umetanje grafikona, premještanje grafikona, dodavanje podataka, umetanje naslova, osi, legende, natpisa nad podatcima, ...
#### N: Kako umetnuti grafikon?

U: Odaberemo željene ćelije u tablici, odaberemo karticu umetanje i u odjeljku grafikoni odaberemo ikonu željenog tipa grafikona.

#### N: Kako premjestiti grafikon u novi list?

U: Desnom tipkom miša prvo odaberemo područje grafikona, a zatim u skočnom izborniku odaberemo element premjesti grafikon. U dijaloškom okviru *Premjesti grafikon* odaberemo gdje ga želimo premjestiti.

#### N: Kako dodati podatke izvan tablice, a da budu prikazani u grafikonu?

U: Upišemo podatke izvan tablice, označimo željene ćelije s podatcima i kopiramo ih (ctrl + C), a zatim odaberemo područje odabira grafikona i zalijepimo podatke u grafikon (ctrl + V).

Minutu prije zvona napomena učenicima da pospreme radna mjesta.

N: Kada zvoni, molim vas pospremite radna mjesta prije nego što izađete iz učionice

Pozdrav učenicima.

N: Hvala vam na suradnji i doviđenja.

Kraj pripreme

# 4.Individualna predavanja-osnovne škole

Nakon četiri upješne pripreme javnih predavanja, došli smo do priprema za individualno predavanje. Podsjećam da na individualnim nastupima nastavnik treba biti puni više angažiraniji nego na javnim predavanjima gdje isti ima podršku od strane metodičara i kolega. Ovo poglavlje donosi dvije pripreme iznimno kvalitetno odrađenih individualnih nastupa.

#### 4.1. Primjer pisane pripreme 1. –individualno predavanje

Individualno predavanje odrađeno je na **Šestoj osnovnoj školi Varaždin**. Tema je bila vrlo specifična i može se reći da je trebalo uložiti dosta osobnog angažmana u primišljanju i planiranju nastavnog sata. Naime tema sata je bila Primjena programiranja u nastavi kemije i fizike. Dakle trebalo je programmiranje u programskom jeziku Logo konkretno i korisno povezati sa sadržajima iz fizike i kemije.

#### Početak pripreme:

Škola: VI. Osnovna škola, Varaždin

Razred: 8. razred

Nadnevak izvođenja nastavnog sata: 15.12.2014.

Nastavna cjelina: Primjena programiranja u nastavi fizike i kemije

Tip sata: obrada novog gradiva

#### Cilj nastavnog sata:

Upoznati i utvrditi korištenje FMS Logo programiranja u fizici .

#### Materijalni zadaci:

- Razumijeti naredbe FMS Logo programskog jezika
- Povezivati programiranje sa sadržajima fizike

#### Funkcionalni zadaci:

- Razvijati dobru orijentaciju unutar FMS Logo okruženja
- Poticati rješavanje fizikalnih zadataka pomoću FMS Logo programskog okruženja
- Programski interpretirati zadatak i rješenje fizikalnog problemskog zadatka
- Poticati interes za samostalno otkrivanje ostalih mogućnosti FMS Logo
- Poticati samostalno izrađivanje zadataka

#### Odgojni zadaci:

- Stjecati pozitivan stav prema radu
- Razvijati suradnju i međusobno pomaganje
- Razvijati samostalnost

#### Nastavna sredstva i pomagala:

- Računalo
- Projektor

#### Nastavne metode:

- Metoda razgovora
- Metoda usmenog izlaganja
- Metoda demonstracije
- Metoda praktičnog rada

#### **Oblici rada:**

- Frontalni rad
- Samostalni rad

#### Literatura:

MOJ PORTAL 3.0, 7 udžbenik informatike s višemedijskim nastavnim materijalima u osmom razredu osnovne škole, Autori: Magdalena Babić, Zoran Dimovski, Fredi Glavan, Mario Stančić, Branko Vejnović; Nakladnik: ŠK-poglavlja 2.5. "Primjena programiranja u nastavi fizike i kemije"

REDNI FAZE DADA LGADDŽAL		METODIČKO	VRIJEME U
BROJ	FAZE KADA I SADKZAJ	OBLIKOVANJE	MINUTAMA
1.	<ul> <li>UVODNI DIO</li> <li>Pozdravljanje i predstavljanje</li> <li>Dogovor o načinu rada</li> </ul>	<ul><li>Usmeno izlaganje</li><li>Razgovor</li></ul>	5'
2	SREDIŠNJI DIO -demonstriranje programskih rješenja fizikalnih problemskih zadataka Primjer 1-Ravnoteža poluge Primjer 2- Ravnoteža poluge (drugi dio iste funkcije) Primjer 3-Električna energija Primjer 4-Izrada koordinatnog sustava Primjer 5-Jednoliko ubrzano gibanje -samostalno rješavanje temeljeno na prepisivanju i testiranju programskog koda uz intervenciju nastavnika	<ul> <li>Usmeno izlaganje</li> <li>Razgovor</li> <li>Demonstracija Individualni rad</li> </ul>	37'
3.	<ul><li>ZAVRŠNI DIO</li><li>Ponavljanje</li></ul>	- Razgovor	3'

- Pozdravljanje	

#### 1. UVODNI DIO (5 minuta)

Predstavljam se i dogovaram pravila rada te dajem uputu da isključe sve nepotrebne sadržaje (youtube, igrice i ostale internetske izvore).

N:Dobar dan.

U:Dobar dan.

N:Moje ime je X.X. i danas ću biti vaš nastavnik.

N:Ako imate bilo kakvih pitanja ili želite nešto reći, podignite ruku i pitajte.

N: U fizici ste govorili o momentima sile nad polugom.

N: Koje ste fizikalne veličine spominjali uz pojam poluge?

U:Silu i krak.

N: Tako je. Na krakove poluge djeluju odgovarajuće sile.

N: Koje ste oznake rabili za silu, a koje za krak?

U: Za silu se rabi oznaka F, a za krak k.

N:Odlično.

N: Do sada ste u fizici moment sile pretežito izračunavali ručno na ploči ili u bilježnici. No danas ćemo primjeniti FMS Logo programsko okruženje za rješavanje fizikalnih problemskih zadataka.

N:Inače u fizici se već dugo koriste programska rješenja za izračunavanje ili simuliranje fizikalnih pojava.

N:Uporabom programskih rješenja, data je mogućnost izrađivanja modela nad kojim se provodi višestruko ispitivanje koje bi bilo iznimno skupo i vremenski zahtjevno da se provodi u realnim uvjetima.

N:Prisjetimo se samo slanja rakete u svemir. Prije nego se šalje raketu u svemir, provode se brojni izračuni i simulacije putem programskih jezika.

N:Programiranje samo za sebe nema svrhu ni smisao. Ono mora biti vezano uz neku konkretnu djelatnost ili aktivnost.

N:Danas ćemo primjeniti programiranje u fizici putem FMS Loga.

N: Zamislimo da sudjelujemo na natjecanju iz fizike.

N: Pred nama je zadatak određivanja ravnoteže poluge nad realnim primjerom klackalice.

(Pokazujem klackalicu)

(Primjer se nalazi u LOGO\_IP.pptx, slajd 2)

N:Pobijediti ćemo samo ako uspješno uspostavimo ravnotežu.

#### 2. SREDIŠNJI DIO SATA(trajanje: 37minuta)

N:Pred nama se nalazi klackalica. Naša prednost pred ostalim natjecateljima je ta što znamo izraditi i primijeniti programe za izračunavanje ravnoteže.

N: Imamo naš FMS Logo koji će biti od iznimne pomoći u izračunavanju ravnoteže.

(Primjer se nalazi u LOGO\_IP.pptx, slajd 3)

N:Posvetimo se klackalici.

N: Koje sve veličine imamo na klackalici?

U: F1, F2, k1, k2

N: Tako je. Na klackalici smo prepoznali sile i krakove.

N:Utvrdimo onda kada je poluga u ravnoteži?

U:Kada su sile lijeve i desne strane jednake.

N: Dobro. No ne zaboravimo da je u fizici točnije upotrijebiti pojam moment sile.

N: Kako bi onda glasila tvrdnja da je poluga u ravnoteži?

U: Poluga je u ravnoteži kada je moment sile lijeve strane jednak momentu sile desne strane.

N: Dobro. Tvrdnju možemo izraziti formulom.

(Primjer se nalazi u LOGO\_IP.pptx, slajd 4)

N: Prije početka natjecanja sastavlja se klackalica.

N: Između velikog broja rezervnih dijelova, zagubio se originalan lijevi krak.

(Primjer se nalazi u LOGO\_IP.pptx, slajd 5)

N: Da bi se natjecanje održalo, potrebno je pronaći odgovarajući krak.

N: No majstori su odokativno odredili lijevi krak.

N: To ne mora značiti da je lijevi krak jednak originalnom kraku koji pripada ovoj klackalici.

N: Bit će lakše ako to izračunamo.

N:Koja su tri elementa klackalice poznata?

U:F1, F2, k1.

N: Dobro poznate su dvije sile i duljina jednog kraka.

N: Imamo tri varijable s jednom nepoznanicom.

N:Kako će onda izgledati formula za izračunavanje kraka k1?

U:Moramo pomnožiti veličine desne strane i podijeliti s lijevom.

(Primjer se nalazi u LOGO\_IP.pptx, slajd 6)

N:Dobro sad tu formulu moramo pretvoriti u liniju koda u FMS Logu da bismo što prije izračunali duljinu kraka **k1.** 

N: Otvorimo FMS Logo. Na nekim računalima FMS Logo se nalazi na radnoj površini. Ako FMS Logo nije na radnoj površini, otvorite ga preko **Start->Programi->FMS Logo**.

N: Ima li netko da nije uspio otvoriti?

(U slučaju da ima interveniram, a ostalim učenicima dajem zadatak da razmisle koji progamski konstrukt će koristiti za rješavanje ovog zadatka. U suprotnom dajem uputu da me prate)

N: Kada imamo nepoznanicu, pitamo se: "Ako nemam k1, onda trebam..."

N:Kojem programskom konstruktu u programiranju pripada ovo pitanje?

U:Uvjetovanju.

N: Tako je. Za rješenje ćemo rabiti IF uvjetovanje.

N: Sad nemojte raditi nego pratite.

N: Prvo moramo izraditi funkciju u koju ćemo po potrebi pozivati u glavni program. U programski editor ćemo napisati naziv funkcije s parametrima: **to poluga :k1 :k2 :f1 :f2.** Parametri funkcije su svi elementi poluge (sile i

krakovi).



N: U novi editor za pisanje programskog koda, upisat ćemo IF uvjetovanje.

N: Prisjetimo se kao izgleda deklaracija IF uvjetovanja?

U:Na prvom mjestu je uvjet, a na drugom naredba.

N: Kada se izvršava naredba?

U: Kada je uvjet ispunjen.

N: Dobro. Odmah iza IF se nalazi nepoznanica označena crvenim obrubom. Njoj moramo pridružiti vrijednsot nula, kako bi se mogla aktivirati naredba. Unutar zelenog obruba se nalazi tekstualni ispis

formule koju ćemo upotrijebiti da izračunamo k1. Unutar plavog obruba se nalazi naredba koja će izračunati krak k1 ukoliko je nepoznat.

(Primjer se nalazi u LOGO\_IP.pptx, slajd 7)

N: Potvrdimo s End i upišimo vrijednosti. Pozivamo poluga 0 3 4 5 i potvrdimo s enter.

N: Dakle k1 je nepoznanica i pridružujemo joj vrijednost 0. Poznate su vrijednosti krak k2 iznosi 3 metra, sila na lijevoj strani iznosi 4 njutna, a na desnoj 5 njutna.

to poluga :k1 :k2 :f1 :f2 poluga defined poluga 0 3 4 5 k1 = (f2\*k2)/f1 = 3.75 m

N:Dobili smo sljedeći rezultat.

N: Lijevi krak k1 mora biti dug 3.75 metara, da bi se postigla ravnoteža.

N:Utvrdili smo duljinu lijevog kraka.

#### Primjer za testiranje 1 (trajanje 4 minute)

N: Stvorite funkciju pod nazivom **poluga :k1 :k2 :f1 :f2**. Prepišite sljedeći kod i unesite vrijednosti redom: **0 3 4 5** 

(Kod i vrijednosti se nalaze u LOGO\_IP.pptx, slajd 8)

(Dok rade obilazim razred, interveniram po potrebi i dajem individualne i kolektivne upute. Kada prođu 4 minute još jedanput prezentiram rješenje zadanog primjera)

N: Dobro. Duljinu kraka k1 smo dobili. Iznosi 3.75 metara.

N: Majstorima smo rekli da zamijene postojeći lijevi krak s krakom duljine 3.75 metara.

N: Zbog nove duljine kraka sad nismo sigurni hoćemo li postići ravnotežu ako za silu f1 unesemo 4 njutna uz uvjet da je krak k2=3 metra, a sila f2=5 njutna.

N:Želimo provjeriti da li smijemo djelovati silom f1 od 4 njutna na novodobiveni krak od 3.75 metara, a da pritom ne bismo narušili ravnotežu poluge.

#### (Primjer se nalazi u LOGO\_IP.pptx, slajd 9)

N:Sad je nepoznanica f1. Kako će izgledati formula za traženje sile f1?

U:Trebamo pomnožiti desnu stranu klackalice i podijeliti s krakom lijeve strane.

N: Tako je. Silu i krak desne strane pomnožimo i podijelimo s duljinom kraka lijeve strane.

(Primjer se nalazi u LOGO\_IP.pptx, slajd 10)

N:Sad prestanite raditi i pratite.

N: U FMS Logu otvorimo pod opcijom **Edall** editor za pisanje programskog koda. Primjetimo da se u njemu nalazi funkcija s napisanom linijom programskog koda za izračunavanje kraka k1.

Editor				۲
e Edit Search Set Test! Help				
poluga :k1 :k2 :f1 :f2 :k1=0 [PR (SE [k1 = (f2*k2)/f1 = ] (:f2*:k2)/:f1 "m)] :f1=0 [PR (SE [f1 = (f2*k2)/k1 = ] (:f2*:k2)/:k1 "N)] d				

N:Ispod te linije upišimo novo uvjetovanje.

(Primjer se nalazi u LOGO\_IP.pptx, slajd 11)

N: Pohranimo kod korištenjem opcije File->Save and Exit.

N: Upišemo u glavni program parametre : poluga 3.75 3 0 5 i potvrdimo s enter.

N: Dobili smo rezultat 4 njutna.

N: Sila f1 je 4 njutna.

N: Dakle lijevi krak možemo opteretiti silom od 4 njutna kao što smo i planirali na početku.

```
poluga 3.75 3 0 5
f1 = (f2*k2)∕k1 = 4 N
```

#### Primjer za testiranje 2(trajanje 4 minute)

N:Prepišite programski kod i unesite sljedeće parametre: 3.75 3 0 5.

(Primjer koda i parametri unosa se nalaze u LOGO\_IP.pptx, slajd 11)

(Dok rade obilazim razred, interveniram po potrebi i dajem individualne i kolektivne upute. Kada prođu 4 minute još jedanput prezentiram rješenje zadatka) N: Sada smo ispitali duljinu kraka i silu na lijevoj strani klackalice.

N: Ako želimo biti doista sigurni u parametre unosa koje smo zadali na početku, moramo provjeriti i desnu stranu.

#### Primjer za testiranje 3 (trajanje 4 minute)

N: Za desnu stranu ćemo upotrijebiti ponovno uvjetovanje samo što ćemo prilagoditi formulu za dobivanje kraka i sile. Umjesto f1 ćemo pisati f2, a umjesto k1 stavimo k2.

N:Prepišite programski kod i unesite sljedeće parametre: 3.75 0 4 5. i 3.75 3 4 0.

(Primjer koda i parametri unosa se nalaze u LOGO\_IP.pptx, slajd 12)

(Dok rade obilazim razred, interveniram po potrebi i dajem individualne i kolektivne upute. Kada prođu 4 minute još jedanput prezentiram rješenje zadatka)

N:Provjerom krakova i sila, utvrdili smo vrijednosti koje će osigurati ravnotežu poluge.

N: Za sada smo sigurni u pobjedu na natjecanju.

N: No pojavio se novi problem.

N:Nestalo je struje. U mraku je teško bilo što raditi.

N: Čuli smo da je nestalo struje zbog pada napona.

N: Natjecanje će se odgoditi ukoliko se za tri sata ne vrati struju.

N:Mjesto gdje se provodi natjecanje mora biti dobro osvijetljeno što znači da je potrebno odrediti točnu vrijednost napona.

N:Dakle, struju je moguće vratiti samo ako se odredi točna veličina napona.

N: Majstori su malo zakazali po pitanju određivanja napona.

N:Opet ćemo koristiti znanje iz fizike i pomoću FMS Loga brzo izračunati kolika će biti vrijednost napona da se ponovno uključi svijetlo.

N: Kako bismo izračunali električnu energiju, koje veličine pritom trebamo uključiti?

U: Napon i vrijeme.

N:Tako je. Potreban je napon i vrijeme.

N: Dakle el. energija je jednaka umnošku napona i vremena.

N:Kako će izgledati onda formula za izračunavanje napona?

U: A=N\*t.

N: Odlično.

(Primjer se nalazi u LOGO\_IP.pptx, slajd 13)

N: Da bismo izračunali napon, koju računsku operaciju ćemo upotrijebiti?

U:Dijeljenje.

N:Dobro. Koje veličine ćemo podijeliti?

U: Energiju i vrijeme.

N: Tako je. Napon je jednak količniku energije i vremena.

(Primjer se nalazi u LOGO\_IP.pptx, slajd 14)

#### Primjer za testiranje 4(trajanje 4 minute)

N: Stvorit ćemo funkciju pod nazivom **struja** s parametrima **:A :N :t.** Koristeći uvjetovanje pronaći ćemo napon struje u Kw.

(Primjer koda se nalazi u LOGO\_IP.pptx, slajd 15)

File	Edit	Search	Set	Test	: Help	)						
to IF IF end	polug :k1=0 :f1=0	a :k1 [PR [PR	:k2 (SE (SE	:f1 [k1 [f1	:f2 = (f2 = (f2	*k2 *k2	2)∕f1 2)∕k1	= =	] (:f2ª ] (:f2ª	€:k2 €:k2	)∕:f1 )∕:k1	"m)] "N)]
to IF end	struj :N=0	a :À [PR (\$	:N : 3E [	t N =	(À∕t)	=	](:A	/:t	) "k₩);	]		

N:Prepišite programski kod i testirajte sljedećim parametrima: t=3 h i A = 450 kWh.

(Primjer koda i parametri unosa se nalaze u LOGO\_IP.pptx, slajd 15)

(Dok rade obilazim razred, interveniram po potrebi i dajem individualne i kolektivne upute. Kada prođu 4 minute još jedanput prezentiram rješenje zadatka).

N: Dakle trebalo je u funkciji napisati novo uvjetovanje koje uključuje formulu za dobivanje napona koja je bila prikazana na prezentaciji.

N: Dobili smo da napon mora biti **150 kW** ukoliko želimo opet dobiti svijetlo.

N: Utvrdili smo točan napon struje. Sada samo trebamo stići do majstora i prenijeti mu tu informaciju.

N: Trebamo odrediti brzinu kojom ćemo prevaliti put.

N: FMS Logo pretežito služi za crtanje.

N:Do sada smo samo računali. No sljedeće računanje ćemo prikazati na grafu.

N: Naime majstori se nalaze na udaljenosti od 25 metara od nas. Vrijeme istjeće i imamo točno 10 sekundi raspoloživog vremena, a da se natjecanje ne prekine.

N:U međuvremeno smo saznali za prečicu duljine 9 metara kojom možemo stići do majstora. Prečicom ćemo se gibati jednoliko.

N: Koristit ćemo s/t graf da bismo grafički vizualizirali jednoliko gibanje kojim ćemo doći do majstora radi lakše orijentacije.

N: Koju veličinu moramo odrediti da bismo stigli do majstora?

U: Koliko brzo idemo.

N: Dobro. Odredit ćemo brzinu kretanja od ishodišta (mjesta gdje sada stojimo ) do odredišta (do udaljene lokacije majstora)

N: Koju ćemu formulu rabiti za određivanje prijeđenog puta, a koristi se u fizici?

U: Put je jednak umnošku brzina i vremenu.

N: Dobro.

#### (Formula se nalazi u LOGO\_IP.pptx, slajd 16)

N: Nas zanima brzina. Za izračunavanje brzine moramo promijeniti postojeću formulu za određivanje prijeđenog puta.

N: Čemu je jednaka brzina?(dajem uputu da gledaju na LOGO\_IP.pptx, slajd 16)

U: Brzina je jednaka razlici prijeđenog puta i vremena.

N: Odlično.

N: Prije nego izračunamo brzinu kretanja, iscrtat ćemo koordinatni sustav radi boljeg pregleda nad vremenom i prijeđenim putem.

N: Od čega se sastoji koordinatni sustav?

U:Od osi x i y.

N: Sastoji se od osi x i y. Što još nedostaje?

U:Mjesto gdje se spajaju x i y.

N:Kako to mjesto zovemo?

U:Ishodište.

N: Tako je. Mjesto iz kojeg izlaze osi x i y zove se ishodište.

N:U FMS Logu ćemo napraviti koordinatni sustav u kojem ćemo dobiti grafički prikaz jednolikog gibanja.

N: Sada pogledajmo program.

(Objašnjavam kod u LOGO\_IP.pptx, slajd 18)

N: Za iscrtavanje koordinatnog sustava u FMS Logu, prvo moramo podići olovku tako da ne ostavlja trag po sučelju za crtanje te da se smjesti uz lijevi krajnji rub sučelja kako bi se osigurao prostor za crtanje koordinatnog sustava . Za taj dio koda zovemo **Pozicioniranje u sučelju za crtanje i** određujemo ishodište koordinatnog sustava.(*Demonstriram slajdovima 19-22 u LOGO\_IP.pptx*) Nakon što smo odredili položaj olovke za crtanje i ishodište, određujemo duljinu osi Y. (*Demonstriram slajdovima 23-25 u LOGO\_IP.pptx* Kada smo nacrtali os Y, vraćamo se natrag na ishodište, kako bismo odredili duljinu osi X. (*Demonstriram slajdovima 26-27 u LOGO\_IP.pptx*) I posljednje je rotacija ispisanog sadržaja na sučelju za crtanje. (*Demonstriram slajdovima 29-29 u LOGO\_IP.pptx*.

#### Primjer za testiranje 5 (trajanje 6 minuta)

N: Otvorite programski editor i u njemu imenujte funkciju **koord :x :y** i prepišite kod za iscrtavanje koordinatnog sustava.

N: Vrijednosti za testiranje su koord 8 80.

(Kod se nalazi u LOGO\_IP.pptx, slajd 30)

Dok rade obilazim razred, interveniram po potrebi i dajem individualne i kolektivne upute. Kada prođu 6 minute sam demonstriram rješenje programa, dajući uputu učenicima da prate.)



N: Sada konačno možemo izračunati brzinu jednolikog gibanja.

N: Pogledajmo programski kod S/T grafa.

#### (Objašnjavam kod u LOGO\_IP.pptx, slajd 31)

N: Naredba **Window** ograničava crtanje na trenutačni zaslon na kojem se nalazi položena olovka za crtanje. Pozivajući koord pozivamo malo prije kreiranu funkciju za iscrtavanje koordinatnog sustava. Samo što umjesto vrijednosti koje smo unosili u koordinatni sustav kao X i Y vrijeddosti ispisuje oznake i njihove mjerne jedinice T(S) S(M). Naredbe MAKE "x -100 + :T\*30, MAKE "y -200 + :S/:T\*3 znače da na Y osi jedinična mjera iznosi 3 piksela, na X osi 30 piksela. Naredba SETXY :x :y iscrtava na temelju unešenih parametar za x i y os. I na kraju Label je izvršna naredba za slučaj da je postavljeni uvjet neistinit što znači da u slučaju neispunjavanja uvjeta ispisuje grešku.

#### Primjer za testiranje 4 (trajanje 7 minuta)

N:Sad nemojte raditi nego gledajte.

N: U programskom editoru nazovemo funkciju **stgraf :S :T.** (*Prikazujem porgramski kod )u* **LOGO\_IP.pptx,** *slajd 30 i objašnjavam naznačene dijelove. Nakon toga demonstriram programski kod sa slajda 31*).

N: Testiranje ćemo napraviti s vrijednostima S=9 metara i T=10 sekundi odnosno stgraf 9 10.



N: Brzina kojom se trebamo kretati da majstorima damo podatak je 0.9 m/s. Moramo se kretati dosta brzo da stignemo do majstora i kažemo koliki je napon potreban da se ponovno uključi struja.

N: Koliko smo rekli da je napon struje?

U: 150 kW.

N:Točno.

N: Stigli smo do majstora i predali stočnu informaciju o naponu struje, svijetlo je proradilo i uz prethodno provjerene parametre vezane za ravnotežu poluge osigurali smo točne razmjere sila i duljine krakova da odredimo ravnotežu poluge.

N:Pobijedili smo na natjecanju zahvaljujući računalnom programu za programiranje u fizici i znanju fizike.

#### 3. ZAVRŠNIO DIO (trajanje: 3 minute)

#### \*PONAVLJANJE.

N:Zatvorite FMS Logo i pospremite radna mjesta.

N: Na današnjem smo satu vidjeli kako programiranje možemo primijeniti u fizici u rješavanju zadataka.

N:Ponovimo.

N:Kod izračunavanja ravnoteže poluge koji smo programski konstrukt koristili?

U:Uvjetovanje.

N:Tako je. Kako je deklarirano uvjetovanje?

U:U prvom dijelu uvjetovanja se postavlja uvjet, a u zadnjem dijelu se postavlja naredba.

N: Kako nazivamo programsku strukturu u koji smo upisivali Logo naredbe.

U: Nazvali smo je funkcija.

N:Tako je. Koristili smo funkcije i pozive funkcija.

N: Pomoću koje naredbe kreiramo funkcije?

U: Pomoću naredbe TO.

N:Dobro.

N: Kako se pozivaju funkcije u FMS Logu?

U: U glavni program upišemo ime funkcije i njezine zadane vrijednosti.

N: Tako je.

N: Na ovom satu smo primjenjivali FMS Logo u fizici, a sljedeći sat ćete se upoznati s mogućnostima korištenja FMS Loga u kemiji.

N: Hvala na suradnji. Bili ste jako marljivi.

N:Možete ići.

N:Doviđenja.

U: Doviđenja.

Zvono.

Kraj pripreme

#### 4.2. Primjer pisane pripreme 2. –individualno predavanje

Drugo individualno predavanje, odnosilo se na programiranje u programskom jeziku **Qbasic na Drugoj osnovnoj školi Varaždin.** Bilo je potrebno odraditi potprogram kao programsku strukturu. U odnosu na prethodno individualno predavanje ovo se doima znanto lakše jer se u programiranje u uključuju sadržaji iz nekih drugih predmeta.

Početak pripreme:

Studijski profil: Informatika u obrazovanju

JMBAG: 0016054629

Škola – vježbaonica: II OŠ Varaždin

Mentor: dr. sc. Goran Hajdin, Darko Krznar, prof.

Datum: 20.11.2014.

Razredni odjel: 8.c

Redni broj nastavnog sata: 17. i 18.

Nastavna cjelina /tema: Programiranje QBasic

Nastavna jedinica: Potprogrami

#### 3. Cilj nastavnog sata:

Definirati i prepoznati potprogram, izraditi program sa potprogramima

#### 4. Zadaće nastavnog sata:

d) Obrazovne zadaće

Učenici će:

- definirati što su potprogrami
- prepoznati strukturu potprograma
- izraditi program sa potprogramima

#### e) <u>Funkcionalne zadaće</u>

Unaprijediti vještine:

- rukovanja računalom
- preciznog izvođenja naredbi
- razvijati sposobnost samostalnog rješavanja zadataka
- razvijati sposobnost povezivanja, zaključivanja i uspoređivanja

#### f) Odgojne zadaće

Poticati učenike:

- samostalnost pri radu
- razvijanje osjećaja osobne vrijednosti
- međusobnu suradnju i pomaganje
- poštivanje utvrđenih pravila komunikacije

#### 3. Tip nastavnog sata

Obrada novog sadržaja

#### 4. Nastavni oblici

Frontalna nastava, individualni rad

#### 5. Nastavne metode

Metoda razgovora, metoda demonstracije, metoda praktičnih radova

#### 6. Nastavna sredstva

računalo, LCD projektor, radni listići

#### 7. Izvori znanja

- Babić M., Dimovski Z., Glavan F., Leko S., Stančić M., Vejnović B. (2014.) «Moj portal», Grafički zavod Hrvatske

- Seršić S. (2006) "Zbirka riješenih zadataka iz programiranja QBasic, Pascal", Pentium, Vinkovci

#### Organizacija nastavnog rada – artikulacija metodičke jedinice

Redni	Faze rada i sadržaj	Metodičko oblikovanje	Vrijeme u minutama
broj			
1.	Uvodni dio	Razgovor	10
	Predstavljanje	Frontalni rad	
	Ponavljanje	Individualni rad	
	Samostalni rad		
2.	Središnji dio	Frontalni rad	75
	Obrada novog sadržaja:	Demonstracija	
	-Potprogrami	Razgovor	
	Demonstracija primjera	Individualni rad	
	Vođena vježba		
	Samostalni rad		
3.	Završni dio		
	- Ponavljanje	Razgovor	5
		Individualni rad	

#### UVODNI DIO SATA (10 min)

Pozdravim učenike i predstavim se.

- N: Dobar dan!
- U: Dobar dan!

N: Ja sam X.X. i održat ću vam ovaj sat, nadam se da ćemo dobro surađivati. Molim vas ako imate kakvih pitanja tijekom sata ili znate odgovor na postavljeno pitanje da dignete ruku.

Danas ćemo raditi programiranje u QBasic-u, kako vi programiranje radite od petog razreda na početku ćemo malo ponoviti, nemojte još otvarati program.

N: Kojom naredbom počinjemo program u QBasic-u?

U: CLS

N: Koju naredbu koristimo za unos podataka?

U: INPUT

N: Koju naredbu koristimo za ispis podataka?

U: PRINT

N: Koji nastavak prilikom spremanja imaju programi u Basicu?

U: bas

N: Sad možete pokrenuti program QBasic.

U: Pokreću program

N: Učenicima dijelim radne listiće,

N: Za ponavljanje izraditi ćemo program koji traži unos dva broja (a i b) i ispisuje ih. Pročitajte na svom radnom listiću zadatak 1.

N: Koje ćemo naredbe koristiti prilikom izrade ovog programa?

U: CLS, INPUT, PRINT

N: Molim vas samostalno riješite zadatak 1. sa radnog listića. Vrijeme za izradu zadatka je 3 min.

Obilazim učenike, odgovaram na postavljena pitanja, provjeravam tko nije završio zadatak.

N: Tko će nam objasniti kako je izradio ovaj program?

U: Učenik čita odgovor liniju po liniju i objašnjava ga, nastavnik ponavlja točan odgovor, pokrećem Slajd 1. na kojem se nalazi izrađen program.

N: Program ćemo spremiti u vašu mapu (ako je imaju) ili radnu površinu pod nazivom Racun.bas

#### SREDIŠNJI DIO SATA (75 min)

Najava teme: Danas ćemo učiti potprograme.

Nastavnik prikazuje naslov: Potprogrami (Slajd: 2)

N: Kada se pripremate za sutrašnji dan u školi, da li istovremeno učite matematiku, hrvatski, pišete zadaću, spremate patike za tjelesni ili rješavate zadatke iz svakog predmeta jedan po jedan?

U: Rješavamo ih jedan po jedan.

N: Kada se gradi kuća da li sve poslove na kući rade isti radnici?

U: Ne, zidaju zidari, vodu rade vodoinstalateri, struju električari ..

N: Ako zamislimo izgradnju kuće kao jedan glavni program, onda sve ostale poslove koje izvode različiti obrtnici koje pozivamo po potrebi da izvrše određene zadatke možemo zamisliti kao potprograme.

Programski jezici imaju sličan način rada, u njima postoje posebni dijelovi potprogrami koji kad zatreba obavljaju samo određen dio zadatka.

- Dijelove programa koji su zaduženi za provedbu nekog zadatka nazivamo potprogrami. Oni su dio glavnog programa. Koristimo ih da bi rješenje programa bilo preglednije i jasnije. (Slajd 3,4)
- N: Sad ćemo naš program prepraviti tako da mu dodamo potprogram koji će zbrojiti unesene brojeve a i b.
- Molim vas sad samo pratite nemojte ništa raditi u svom programu, na primjeru pokazat ću izradu potprograma za zbrajanje. (Slajd 5)

N:

- Na traci izbornika odaberemo izbornik Edit u kojem odaberemo naredbu: New SUB. (Slajd: 6) Otvorit će nam se novi prozor. U okvir Name: upišemo naziv potprograma, Zbroj, potvrdimo odabirom naredbe OK. (Slajd: 7)
- U prozoru programa pokazat će nam se novi prozor u koji je već upisano:
- SUB Zbroj
- END SUB (Slajd: 8) (Trajanje demonstracije: 4 minute)

Svaki program počinje ključnom riječi SUB koja je skraćenica od engleske riječi Subrutine, a završava naredbom END SUB. Između tih linija upisujemo naredbe potprograma.

N: Nazivu potprograma Zbroj dodajemo varijable koje ćemo koristiti

N: Koje su to varijable?

U: a i b

- N: Koju naredbu koristimo za ispis zbroja dva broja?
- U: PRINT «Zbroj brojeva»;a;»+»;b;»=»;a+b

#### upisujemo naredbe potprograma: PRINT «Zbroj brojeva»;a;»+»;b;»=»;a+b (Slajd: 9)

Naš glavni program ostao je sačuvan, odabirom tipke F2 otvara nam se prozor u kojem vidimo naziv glavnog programa i potprograma koji su uvučeni ispod glavnog programa, klikom na naziv glavnog programa vraćamo se u njega i upisujemo naziv potprograma na mjestu gdje želimo da se potprogram izvrši. (Slajd:10)

Upisom: Zbroj a, b – pozivamo izvršenje našeg potprograma. Program spremimo naredbom Save. Pokrenemo ga tipkom F5. (Slajd :11, 12)

N: Sad ćemo testirati program Racun.bas upisom brojeva 2 i 5. (Slajd:13)

N: Sad ćemo zajedno u programu Racun.bas izraditi ovaj potprogram. Vračam se na (Slajd 5) i vodim učenike korak po korak kroz izradu potprograma za zbrajanje. (Trajanje: 4 minute)

N: Tko nije uspio izvršiti zadatak. Pomažem učenicima koji nisu uspjeli.

N: Svaki program može imati veći broj potprograma, pokušajte samostalno na ovom istom programu dodati potprogram: Razlika koji će od broja a oduzeti broj b.

Učenicima dajem im upute za rad: Potrebno je samostalno izraditi zadatak 2. na radnom listiću, prozivam jednog učenika da kaže postupak kojim ćemo riješiti zadatak. Ponavljam postupak. Vrijeme za izradu zadatka je 3 min učenici samostalno rješavaju zadatak, obilazim učenike i po potrebi im pomažem u radu.

Provjeravam tko nije završio zadatak?

Prozivam jednog učenika da objasni kako je riješio zadatak, rješenje prikazujem na (Slajdu:14).

N: Na koji smo način pozvali potprogram iz glavnog programa?

U: Upisujući ime potprograma u glavni program.

N: Potprograme možemo pozvati na još jedan način, upotrebom naredbe CALL na taj način stvaramo bolju preglednost programa.

Molim vas da sada pratite pokazat ću vam pozivanje potprograma naredbom CALL, nakon toga zajedno ćemo probati dodati vašem programu još jedan potprogram. Demonstriram učenicima na istom programu izradu potprograma Umnožak upotrebom naredbe CALL. (Slajd 15-18)

Sad ćemo u programu Racun.bas napraviti potprogram Umnožak.

N: Tko će nas voditi kroz vježbu i objasniti postupak.

U: Učenik koji se javi izvodi vježbu i verbalno prati svoje postupke sa svoga radnog mjesta, po potrebi pojašnjavam i govorim postupke.

N: Riješiti ćemo sad zadatak 3 na radnom listiću. Svome programu Račun dodajte samostalno upotrebom naredbe CALL novi potprogram: Kolicnik. Vrijeme izrade zadatka je 3 min.

Obilazim učenike i provjeravam tko nije završio zadatak?

N: Program ćemo spremiti u vašu mapu (ako je imaju) ili radnu površinu pod nazivom Racun1.bas (Slajd 19)

N: Sada će zvoniti, ovaj zadatak završiti ćemo sljedeći sat.

Mali odmor

N: Nastavit ćemo rad na programu Racun.bas. Što će se dogoditi ako u ovom našem programu pozovemo još jednom potprogram zbroj?

U: Potprogram zbroj će se još jednom izvesti.

N: Iz glavnog programa, ako za to postoji potreba možemo potprograme pozvati više puta. U ovom programu pozvat ćemo iza potprograma: Količnik još jednom potprogram: Zbroj. Tko će nam demonstrirati vježbu, ostali učenici paralelno rade na svom računalu.

U: Učenik se javlja i demonstrira na svom računalu, ja ga verbalno pratim.

N: Provjerava tko nije uspio riješiti zadatak, pomaže mu u radu. (Slajd 20, 21)

N: Program spremite naredbom Save i zatvorite.

N: Sad ćemo izraditi program koji od korisnika traži da odabere želi li izračun kvadrata ili kuba upisanog broja. Ovisno o odabiru korisnika izračunat će kvadrat ili kub. Po izvođenju program treba izgledati ovako. (Slajd 22)

N: Koju naredbu koristimo ako u programu imamo uvjet?

U: IF THEN

N: Koliko koraka ima ovaj program?

U: 5

N: Koji je prvi korak programa?

U: Program će ispisati "Program koji računa kvadrat ili kub upisanog broja"

N: Što program od nas zahtjeva u sljedećoj liniji?

U: Upis broja

N: Što program zahtjeva u trećoj liniji?

U: Da odaberemo što želimo računati.

N: Nakon toga ovisno o našem odabiru, program će izračunati kvadrat ili kub. (Slajd 23)

N. Sad ćemo zajedno izraditi ovaj program. Nastavnik vodi učenike korak po korak, učenici na vlastitim računalima izrađuju program i potprogram kvadrat. Slajdovi (24-26)

N: Potprogram ćemo pozvati naredbom IF THEN.

Provjeravam tko nije uspio završiti zadatak, program spremamo u njihovu mapu pod nazivom: kvadr\_kub.bas, ako su učenici završili zadatak dajem im uputu za samostalni rad.

N: Zadatak je u ovom programu samostalno izraditi potprogram kub (Zadatak 4 u radnom listiću). Vrijeme za izradu zadatka je 5 min.

Učenici samostalno rade, obilazim ih i pomažem u radu, učenici koji prije završe mogu pomoći učenicima koji su sporiji ili trebaju pomoć. Analiziramo zadatak (Slajd 27) Provjeravam tko nije završio zadatak, ako su svi gotovi upućujem ih na zadatak u radnom listiću.

#### N: Nabrojite korake izrade ovoga programa? (Slajd 28)

U: Učenik govori korake izrade programa. (Slajd 29)

N: Samostalno izradite zadatak 5 u radnom listiću koristeći se pri tome naredbama i postupcima koje smo koristili prilikom izrade prethodnog programa. Na platnu se nalazi rješenje prethodnog zadatka koje vam može pomoći prilikom izrade (Slajd 27).

Učenici samostalno rade, obilazim ih i pomažem u radu, učenici koji prije završe mogu pomoći učenicima koji su sporiji ili trebaju pomoć. Provjeravam da li su svi završili zadatak, učenicima dijelim radni listić 2. na kojem se nalaze rješenja svih programa koje smo danas radili da bi ih mogli zalijepiti u bilježnicu. (Slajd 30, 31)

#### Dodatni zadatak:

U slučaju da učenici ranije završe zadatak, upućujem ih da u preprave prvi izrađeni program Racun.bas kako bi korisnik mogao odabrati želi li izračun zbroja, razlike, umnoška ili količnika.

#### ZAVRŠNI DIO SATA (5 min)

Dajem učenicima uputu da isključe računala, i postavljam nekoliko pitanja za ponavljanje.

- N: Što su potprogrami?
- U: Posebni dijelovi programa zaduženi za provedbu nekog zadatka.
- N: Kojom naredbom počinje svaki potprogram?
- U: SUB
- N: Kojom naredbom završava potprogram?
- U: End sub
- N: Na koji način pozivamo potprograme?
- U: Naredbom CALL i navođenjem imena potprograma u glavom programu.
- N: Što će se dogoditi ako u editoru QBasic-a pritisnemo tipku F2.
- U: Otvorit će nam se prozor sa strukturom programa.

N: Na radnom listiću nalazi se križaljka sa pojmovima iz današnje lekcije, u polje koje počinje npr. brojem 1 upišite odgovor na prvo pitanje...

Provjeravam tko nije uspio završiti križaljku, prozivam učenike koji se jave te oni čitaju odgovore iz križaljke. (Slajd: 31)

N: Zahvaljujem se na suradnji, uskoro će zvoniti, spremite svoja radna mjesta. Doviđenja.

#### Radni listić 1.

#### Zadatak 1.

Izradi program koji traži unos dva broja (a i b) i ispisuje ih. Pokretanjem programa trebamo dobiti ovaj upit. (Vrijeme predviđeno za izradu zadatka: 3 min.)



#### Zadatak 2.

U programu **Racun.bas** izradi potprogram **Razlika** koji će od broja a oduzeti broj b. (Vrijeme predviđeno za izradu zadatka: 3 min.)

#### Zadatak 3.

Korištenjem naredbe CALL u programu **Racun.bas** izradi potprogram **Kolicnik** koji će broj a podijeliti sa krojem b. (Vrijeme predviđeno za izradu zadatka: 3 min.)

#### Zadatak 4.

U programu **Kvadr\_kub.bas** izradi potprogram **Kub** koji će biti pozvan ukoliko korisnik odabere broj 3.

#### Zadatak 5.

Upotrebljavajući naredbe koje smo koristili prilikom izrade programa Kvadr\_kub.bas izradi program **Geom\_lik.bas** koji ovisno o odabiru korisnika računa: opseg i površinu kvadrata, pravokutnika ili pravokutnog trokuta.

Program koji racuna opseg i povrsinu zadanog lika. Odaberi geometrijski lik s popisa: 1 Kvadrat 2 Pravokutnik 3 Pravokutan trokut Upisi broj geometrijskog lika:

#### Zadatak 6.

Odgovarajući na postavljena pitanja riješi križaljku.

- 1. Kako se naziva programski jezik u kojem smo danas radili?
- 2. Na kojem tekstualnom izborniku odabiremo naredbu New Sub?
- 3. Kojom ključnom riječi počinje potprogram?
- 4. Kojom naredbom pozivamo potprograme u QBasic-u?
- 5. Kako se nazivaju posebni dijelovi programa zaduženi za provedbu nekog zadatka?

#### Radni listić 2.

- 1. Program koji računa zbroj, razliku, umnožak i količnik dva unesena broja.
  - INPUT "Upisite brojeve a i b:"; a, b Zbroj a, b razlika a, b CALL umnozak(a, b)



```
CALL kolicnik(a, b)
```

#### PRINT

SUB Zbroj (a, b) PRINT "Zbroj brojeva"; a; "+"; b; "="; a + b END SUB SUB razlika (a, b)

PRINT "Razlika brojeva"; a; "-"; b; "="; a - b

END SUB

SUB umnozak (a, b)

PRINT "Umnozak brojeva"; a; "\*"; b; "="; a \* b

#### END SUB

SUB kolicnik (a, b) PRINT "Kolicnik brojeva"; a; "/"; b; "="; a / b END SUB

#### 2. Program koji ovisno o odabiru korisnika računa kvadrat ili kub upisanog broja

PRINT "Program koji racuna kvadrat ili kub upisanog broja." PRINT INPUT "Za koji broj zelis izracunati kvadrat ili kub: ", a

INPUT "Za izracun kvadrata, odaberi 2, a za izracun kuba, odaberi 3. ", o

IF o = 2 THEN CALL kvadrat(a)

IF o = 3 THEN CALL kub(a)

SUB kvadrat (a)

PRINT

PRINT "Kvadrat upisanoga broja je"; a \* a

END SUB

SUB kub (a) PRINT PRINT "Kub upisanoga broja je"; a \* a \* a END SUB

3. Program koji ovisno o odabiru korisnika računa opseg i površinu kvadrata, pravokutnika ili pravokutnog trokuta.

PRINT "Program koji racuna opseg i povrsinu zadanog lika."

PRINT

PRINT "Odaberi geometrijski lik s popisa:"

PRINT "1 Kvadrat"

PRINT "2 Pravokutnik"

PRINT "3 Pravokutan trokut"

INPUT "Upisi broj geometrijskog lika: ", b

IF b = 1 THEN CALL Kvadrat(a)

IF b = 2 THEN CALL Pravokutnik(a, b)

#### IF b = 3 THEN CALL Trokut(a, b, c)

SUB Kvadrat (a)

INPUT "Upisi duljinu stranice a: ", a PRINT "Povrsina kvadrata je:"; a \* a

PRINT "Opseg kvadrata je:"; 4 \* a

END SUB

SUB Pravokutnik (a, b)

INPUT "Upisi duljinu stranice a: ", a

INPUT "Upisi duljinu stranice b: ", b

PRINT "Opseg pravokutnika je:", 2 \* (a + b)

PRINT "Povrsina pravokutnika je:", a \* b

END SUB

SUB Trokut (a, b, c)
INPUT "Upisi duljnu stranice a: ", a
INPUT "Upisi duljnu stranice b: ", b
INPUT "Upisi duljnu stranice c: ", c
PRINT "Povrsina pravokutnog trokuta je:"; (a \* b) / 2
PRINT "Opseg pravokutnog trokuta je:"; a + b + c
END SUB

Kraj pripreme:

## 5. Primjer pisane pripreme za izvedbu nastavnog sata na stručnom ispitu

Priprema za izvođenje nastavnog sata prilikom polaganja stručnog ispita, zhtijeva poseban obrazac kojeg dodjeljuje mentor na stručnom ispitu. Radi se o puno detaljnijem i usustavljenom obrascu za planiranje i osmišljavanje nastavnog sata kojeg propisuje Agencija za odgoj i obrazovanje.

U nastavku možete vidjeti sam prazan obrazac za pisanje pripreme za izvođenje nastavnog sata na stručnom ispitu.

#### 5.1. Primjer obrasca pisane priprave za izvedbu nastavnog sata na stručnom ispitu

U nastavku možete vidjeti sam prazan obrazac za pisanje pripreme za izvođenje nastavnog sata na stručnom ispitu.

### **PISANA PRIPRAVA** ZA IZVEDBU NASTAVNOG SATA NA STRUČNOM ISPITU

Pristupnik:

Mentor na stručnome ispitu:

Škola:

Nadnevak:

#### PRIPREMA ZA IZVOĐENJE NASTAVE

Ško	ola:				Mjesto:
		Nast	avni predme	t:	
		Razred:			
Nas	stavna	cjelina:			Smjer:
Nas	stavna	tema:			Zanimanje:
Pri	stupni	k:			
Dat	tum izv	vođenja oglednog sa	ata:		
1.	Podje 1.1. n 1.1 1.2. n 1.2 1.2 1.3. N	la: astavne cjeline na t l.1. 2. astavne teme na jeo 2.1. 2.2. astavna tema/jedin	teme: dinice: iica:		
2.	Vrsta	metodičke jedinice	(tip sata):		
3.	Obraz	obrada novih nasta ponavljanje vježbanje provjeravanje ovni ishodi:	vnih sadržaja		

#### 4. Ključni pojmovi:

#### 5. Cilj nastavne teme/jedinice:
## 6. Obrazovni ishodi:

Kognitivna (spoznajna – materijalna)	Psihomotorička (funkcionalna)	Afektivna (odgojna)
Učenici će:	Učenici će:	Razvijati kod učenika:
-	-	-

# 7. Oblici rada:

# 8. Nastavne metode:

9. Nastavna sredstva i pomagala:

# 10. Korelacija:

### 11. Literatura i izvori:

a) *za učenike* 

# b) za nastavnika:

# ORGANIZACIJA NASTAVNOG SATA (detaljno razraditi)

Етара	ARTIKULA	CIJA NASTAVNOG	SATA	ETODE	EDSTVA I LA	(NIM
NASTAVNOGA SATA	Sadržaj	ADRŽAJ AKTIVNOST AKTIVNOST NASTAVNIKA UČENIKA		NASTAVNE M	NASTAVNA SRI POMAGA	VRIJEME (
<ol> <li>PRIPREMA</li> <li>1.1. Materijalno – tehnička priprema</li> </ol>						
1.2. SADRŽAJNA PRIPREMA						
1.3. PSIHOLOŠKA priprema učenika (motivacija)						
2. Obrada novih nastavnih sadržaja						
3. Vježbanje						
4. Ponavljanje						
5. Provjeravanje						

# TIJEK IZVOĐENJA NASTAVE

(detaljna razrada – "scenarij" nastavnog procesa)

# PLAN PLOČE

PRILOZI

## 5.2. Pisana priprava za izvedbu nastavnog sata na stručnom ispitu

Tema izvedbe nastavnog sata na stručnom ispitu je bila primjena for petlje u Pythonu. Dakle, učenici su sat prije imali uvodno predavanje u for petlji, gdje će im biti objašnjena sintaksa, neki jednostavni primjeri tipa što program radi te jednostavni primjeri s for petljom.

Dio izvedbenog sata pisane pripreme je bio rješavati neke primjere s for petljom.

Sat je bio organiziran u računalnoj učionici s pola razrednog odjeljenja (do 14 učenika). Ideja je bila pripremiti zadatke od lakših prema težim, primjerice 5 zadataka i onda ih se tijekom sata rješavalo. Prije svakog zadatka s učenicima je detaljno prokomentirano kako riješiti zadatak. Nije preporučljivo imati rješenja primjerice na PowerPointu koja će se samo pokazati učenicima već je potrebno zahtijevati da oni učenici aktivno sudjeluju na satu, da razmišljaju, iznose svoje ideje, makar one bile i skroz krive.

# PISANA PRIPRAVA ZA IZVEDBU NASTAVNOG SATA NA STRUČNOM ISPITU

Pristupnik:

Mentor na stručnome ispitu:

X.X

X.X

Škola: V. gimnazija Zagreb

Nadnevak: 01.03.2017.

# PRIPREMA ZA IZVOĐENJE NASTAVE

Škola: V.gimnazija Zagreb, Klaićeva 1 Mjesto:Zagreb, Klaićeva 1

Nastavni predmet: Informatika Razred: 1f

Nastavna cjelina: Programiranje (Python) Smjer: Program C

Nastavna jedinica: Primjena for petlje Zanimanje: mag.edu.inf.

Pristupnik: Zoran Hercigonja

Datum izvođenja oglednog sata: 01.03.2017.

# 2. Podjela:

- **1.1.** Nastavne cjeline na teme:
  - 2.1. **1.** Uvod u programiranje
  - 1.1.2.Unos i ispis podataka
  - 1.1.3.Naredba grananja
  - 1.1.4.Petlje
- **1.2.** Nastavne teme na jedinice:
  - **1.2.1**.Uvod u for petlju
  - **1.2.2.Primjena for petlje**
  - **1.2.3.**Uvod u while petlju
  - **1.2.4.**Primjena while petlje

# 3. Vrsta metodičke jedinice (tip sata):

□ Vježbanje

# 4. Obrazovni ishodi:

- 🗆 Kognitivni
- Psihomotorički
- Afektivni

### 5. Ključni pojmovi:

□ For petlja, brojač petlje, korak petlje, ponavljanje

# 6. Cilj nastavne teme/jedinice:

- Upoznati učenike s konkretnom primjenom for petlje u programskom jeziku Python
- D Pripremiti učenike za samostalno rješavanje zadataka

## 7. Obrazovni ishodi:

Učenici će: Razvijati kod učenika:	
<ul> <li>-usvojiti algoritam funkcioniranja for petlje</li> <li>- prepoznati situacije u kojima je moguće primijeniti for petlju</li> <li>- povezivati ostale programske strukture sa for petljom</li> <li>- primijeniti for petlju za rješavanje konkretnog problema</li> <li>- rješavati programske zadatke unotrebom for petlje</li> <li>- kreativno rješavati konkretne zadatke upotrebom for petlje</li> <li>- samostalno otkrivati primjenu for petlje u nekom drugom problemskom kontekstu</li> <li>- samostalnost i odgovorn</li> </ul>	radu zaciju ost

# 8. Oblici rada:

- Frontalni rad
- □ Individualan rad

#### 9. Nastavne metode:

- Metoda usmenog izlaganja
- Metoda razgovora
- □ Metoda demonstracije
- Metoda praktičnog rada

### 10. Nastavna sredstva i pomagala:

- 🗌 Računalo
- □ Projektor
- Programski alat Python
- Power Point prezentacija

- Online Python vizualizator
- Nastavni listići
- Bijela ploča

### 11. Korelacija:

- □ Mobilne aplikacije za pametno obavještavanje
- □ Konkretan primjer iz svakodnevnog života koji je učenicima najbliži, a ujedno omogućava lakše razumijevanje sadržaja nastavnog sata

#### 12. Literatura i izvori:

#### a) za učenike

□ Znanje.org-riješeni primjeri i zadaci u Pythonu http://www.znanje.org/knjige/computer/python/python\_abc\_zadaci.htm

#### b) za nastavnika:

- □ L. Budin, P. Brođanac, Z.Markučić, S.Perić-Rješavanje problema programiranjem u Pythonu, udžbenik za prirodoslovno-matematičke gimnazije
- □ INFO-LUX zadaci i vježbe iz informatike <u>http://info-lux.net/python-vjezbe-grananja-petlje/</u>

# ORGANIZACIJA NASTAVNOG SATA (detaljno razraditi)

Етара	ARTIKU	JLACIJA NASTAVNOG	SATA	ETODE	BDSTVA I LA	(NIM
NASTAVNOGA SATA SADRŽAJ AKTIVNOS NASTAVNIJ		Aktivnost nastavnika	Aktivnost učenika	NASTAVNE M	NASTAVNA SRI POMAGA	Vrijeme (
6. PRIPREMA 6.1. Materijalno - tehnička priprema	Priprema radnog mjesta učenika	Daje uputu o uključivanju računala	Slijede upute nastavnika. Postavljaju pitanja.	Razgovor, praktične aktivnosti	Računalo	2 minute
6.2. Sadržajna priprema	Priprema prezentacije i programskih primjera	Na nastavničkom računalu otvara nastavne sadržaje Daje uputu o pokretanju Python editora za pisanje programskog koda	Slijede upute nastavnika. Postavljaju pitanja.	Razgovor, praktične aktivnosti	Računalo	1 minuta
6.3. <b>P</b> sihološka priprema učenika (motivacija)	Uvodni motivacijski primjer korištenja for petlje u svakodnevnom životu. Primjer pametnih obavijesti u mobilnim uređajima.	Demonstrira isječak programskog koda. Sustavom potpitanja potiče učenike na interakciju i razgovor. Zadaje učenicima da prepišu i testiraju kod.	Prepisuju i testiraju programski kod na svojim računalima. Samostalno postavljaju pitanja. Odgovaraju na nastavnikova pitanja.	Demonstracija, razgovor, usmeno izlaganje, Praktične aktivnosti	Računalo, projektor, bijela ploča, programski alat Python <u></u> Python editor za pisanje programskog koda	4 minute
7. Vježbanje 2.1. Rješavanje Zadatka 1		Osmišljenim sustavom pitanja i potpitanja zajedno s učenicima konstruira rješenje na ploči. Odgovara na pitanja učenika.	Odgovaraju na pitanja nastavnika. Postavljaju pitanja nastavniku. Sudjeluju u raspravi i konstrukciji rješenja		Računalo,	

	Power Point prezentacija	Prati rad učenika za vrijeme samostalnog rješavanja zadatka. Intervenira po potrebi. Nakon rješavanja zadatka, izvodi rješenje koristeći Python vizualizator Komentira rješenje.	zadatka na ploči. Samostalno rade i programiraju rješenje zadatka. Obraćaju se nastavniku ukoliko naiđu na poteškoću u rješavanju.	Usmeno izlaganje, razgovor, praktične aktivnosti, demonstracija	projektor, bijela ploča, PowerPoint prezentacija, programski alat Python, online Python vizualizator	6 minuta
2.2. Rješavanje Zadatka 2	Power Point prezentacija	Osmišljenim sustavom pitanja i potpitanja zajedno s učenicima konstruira rješenje na ploči. Odgovara na pitanja učenika. Prati rad učenika za vrijeme samostalnog rješavanja zadatka. Intervenira po potrebi. Nakon rješavanja zadatka, izvodi rješenje koristeći Python vizualizator Komentira rješenje.	Odgovaraju na pitanja nastavnika. Postavljaju pitanja nastavniku. Sudjeluju u raspravi i konstrukciji rješenja zadatka na ploči. Samostalno rade i programiraju rješenje zadatka. Obraćaju se nastavniku ukoliko naiđu na poteškoću u rješavanju.	Usmeno izlaganje, razgovor, praktične aktivnosti, demonstracija	Računalo, projektor, bijela ploča, PowerPoint prezentacija, programski alat Python, online Python vizualizator	6 minuta
2.3. Rješavanje Zadatka 3	Power Point prezentacija	Osmišljenim sustavom pitanja i potpitanja zajedno s učenicima konstruira rješenje na ploči. Odgovara na pitanja učenika. Prati rad učenika za vrijeme samostalnog rješavanja zadatka. Intervenira po potrebi. Nakon rješavanja zadatka, izvodi rješenje koristeći Python vizualizator Komentira rješenje.	Odgovaraju na pitanja nastavnika. Postavljaju pitanja nastavniku. Sudjeluju u raspravi i konstrukciji rješenja zadatka na ploči. Samostalno rade i programiraju rješenje zadatka. Obraćaju se nastavniku ukoliko naiđu na poteškoću u	Usmeno izlaganje, razgovor, praktične aktivnosti, demonstracija	Računalo, projektor, bijela ploča, PowerPoint prezentacija, programski alat Python, online Python	7 minuta

			rješavanju.		vizualizator	
2.4. Rješavanje Zadatka 4	Power Point prezentacija	Osmišljenim sustavom pitanja i potpitanja zajedno s učenicima konstruira rješenje na ploči. Zajedno s učenicima raščlanjuje tekst zadatka na manje dijelove i obrazlaže smisao zadatka. Odgovara na pitanja učenika. Prati rad učenika za vrijeme samostalnog rješavanja zadatka. Intervenira po potrebi. Nakon rješavanja zadatka, izvodi rješenje koristeći Python vizualizator	Odgovaraju na pitanja nastavnika. Postavljaju pitanja nastavniku. Sudjeluju u raspravi i konstrukciji rješenja zadatka na ploči. Samostalno rade i programiraju rješenje zadatka. Obraćaju se nastavniku ukoliko naiđu na poteškoću u rješavanju.	Usmeno izlaganje, razgovor, praktične aktivnosti, demonstracija	Računalo, projektor, bijela ploča, PowerPoint prezentacija, programski alat Python, online Python vizualizator	6 minuta
2.5. Rješavanje Zadatka 5	Power Point prezentacija	<ul> <li>pitanja i potpitanja</li> <li>zajedno s učenicima</li> <li>konstruira rješenje na</li> <li>ploči.</li> <li>Zajedno s učencima</li> <li>raščlanjuje tekst zadatka</li> <li>na manje dijelove i</li> <li>obrazlaže smisao zadatka.</li> <li>Odgovara na pitanja</li> <li>učenika.</li> <li>Daje uputu o mogućnosti</li> <li>korištenja dijelova</li> <li>programskog koda iz</li> <li>prethodnog zadatka.</li> <li>Prati rad učenika za</li> <li>vrijeme samostalnog</li> <li>rješavanja zadatka,</li> <li>izvodi rješenje koristeći</li> <li>Python vizualizator</li> </ul>	Odgovaraju na pitanja nastavnika. Postavljaju pitanja nastavniku. Sudjeluju u raspravi i konstrukciji rješenja zadatka na ploči. Samostalno rade i programiraju rješenje zadatka. Obraćaju se nastavniku ukoliko naiđu na poteškoću u rješavanju.	Usmeno izlaganje, razgovor, praktične aktivnosti, demonstracija	Računalo, projektor, bijela ploča, PowerPoint prezentacija, programski alat Python, online Python vizualizator	8 minuta

		Komentira rješenje.				
2.6. Rješavanje Zadatka 6	Power Point prezentacija	Osmišljenim sustavom pitanja i potpitanja zajedno s učenicima konstruira rješenje na ploči. Zajedno s učencima raščlanjuje tekst zadatka na manje dijelove i obrazlaže smisao zadatka. Odgovara na pitanja učenika. Daje uputu o mogućnosti korištenja dijelova programskog koda iz prethodnog zadatka. Prati rad učenika za vrijeme samostalnog rješavanja zadatka. Intervenira po potrebi. Nakon rješavanja zadatka, izvodi rješenje koristeći Python vizualizator Komentira rješenje.	Odgovaraju na pitanja nastavnika. Postavljaju pitanja nastavniku. Sudjeluju u raspravi i konstrukciji rješenja zadatka na ploči. Samostalno rade i programiraju rješenje zadatka. Obraćaju se nastavniku ukoliko naiđu na poteškoću u rješavanju.	Usmeno izlaganje, razgovor, praktične aktivnosti, demonstracija	Računalo, projektor, bijela ploča, PowerPoint prezentacija, programski alat Python, online Python vizualizator	5 minuta
3. ZAVRŠNI DIO	NASTAVNI_LISTIĆ_Domaća _zadaća	Dijeli nastavne listiće s domaćom zadaćom učenicima daje kratke upute za rješavanje zadataka.	Slijede upute i postavljaju pitanja nastavniku.	Razgovor	Nastavni listić	3 minute

# **TIJEK IZVOĐENJA NASTAVE**

(detaljna razrada – "scenarij" nastavnog procesa)

# UVODNI DIO (trajanje do 4 minute)

U uvodnom dijelu sata ću se predstaviti učenicima i dogovoriti način rada. Dogovor oko načina rada podrazumijeva pravilo podizanja ruke ukoliko učenik želi odgovoriti na pitanje, postaviti upit nastavniku ili pozvati nastavnika ukoliko naiđe na problem tijekom rješavanja zadataka.

N: Dobar dan.

U: Dobar dan.

N: Zovem se Zoran Hercigonja i danas ću biti vaš nastavnik.

N: Ukoliko tijekom sata odgovarate na pitanje ili želite postaviti pitanje u vezi s gradivom, molim podignite ruku. Kad vas prozovem pitajte ili odgovorite.

N:Tako ćemo izbjeći nepotrebnu buku i svatko će dobiti jednaku priliku za odgovaranje ili postavljanje pitanja. Jesmo se razumjeli?

U: Jesmo.

N: Pokrenite Python aplikaciju.

# GLAVNI DIO (trajanje do 37 minuta)

Započinjem s motivacijskim primjerom iz svakodnevnog života.

N: Svakodnevno koristite mobilne uređaje.

N:Koristite ih u komunikacijske svrhe te u svrhu pretraživanja interneta.

N: Osim tih mogućnosti, mobilni uređaji i njihove aplikacije imaju mogućnost redovitog obavještavanja o nadogradnjama, ažuriranju, pristizanju obavijesti, ali i mogućnost određivanja podsjetnika za vaše osobne aktivnosti. Sve te obavijesti se ponavljaju.

N: Jeste li ikada koristili kakav oblik podsjetnika na mobilnom uređaju.

U:Jesmo.

#### N: Kakav je to podsjetnik bio?

U: Podsjetnik za odlazak na trening.

N: Podsjetnik za trening je bio tempiran u nekom vremenskom intervalu pretpostavljam. Poruka se ispisivala svaki puta u isto vrijeme baš poput budilice koja zvoni svaki dan u isto vrijeme koje ste unaprijed sami odredili.

Često upotrebljavam riječ ponavljanje kako bih na taj način povezao teorijske postavke for petlje s konkretnu primjenu programske.

#### Zadatak 0-Uvodni\_motivacijski

#### Demonstriram i obrazlažem motivacijski zadatak na ploči i računalu

N: Budući da još uvijek vlada sezona gripe, zamislimo da je liječnik preporučio uzimati vitaminski napitak za jačanje organizma. Kako ne bismo zaboravili uzeti napitak svakog dana, na mobilnom uređaju tempiramo podsjetnik.

N:Aplikacije za pametne obavijesti često u pravilnim intervalima podsjećaju korisnika da mu je pristigla neka obavijest. Na primjer svake dvije minute ili svakih pet sekundi se ponovno pojavljuje obavijest dok korisnik ne reagira i otvori tu poruku.

N: U sljedećem primjeru kreiran je jedan takav program?

#### N: Gdje smo u programu omogućili da se poruka ponavlja određeni broj puta?

U: U intervalu for petlje smo odredili broj ponavljanja.

#### N: Tako je. Koliko puta će se ispisati poruka "Podsjetnik-uzeti napitak"?

U: Poruka će se ispisati 10 puta.

N: Izvrsno. Dakle poruka se ispisuje 10 puta.

#### N: Kako bismo napravili da se poruka ispisuje na primjer 17 puta?

U: U for petlji bismo na mjesto završne vrijednosti upisali 17.

N: Izvrsno. Umjesto 10 bismo napisali 17 i toliko puta bi se poruka ispisala.

N: Osim samog ispisa poruke omogućeno je da se ona ispisuje u nekom vremenskom intervalu.

N: U našem primjeru smo odredili ispisivanje poruke svake dvije sekunde.

#### N: Gdje u kodu određujemo ispis svake dvije sekunde?

- U: Ispis svake dvije sekunde se određuje u varijabli a.
- N: Dobro. Odgovor je djelomično točan.

#### N: Gdje se još u kodu nalazi varijabla a?

U: Varijabla a se nalazi u naredbi *time.sleep(a)*.

N: Točno.

#### N: Znamo li što je imort time?

U: Import time je biblioteka.

N: Tako je. Import time je biblioteka koja se uključuje kako bi bilo moguće raditi s vremenskim funkcijama na primjer za dohvaćanje lokalnog vremena i manipulacije tim vremenom.

#### N: Jeste li dosad koristili kakvu biblioteku?

U: Koristili smo biblioteku import random za rad s nasumičnim brojevima.

N: Izvrsno. Svaka biblioteka ima svoj skup funkcija.

Biblioteka time	import time
Unos vremena ispisa	a=2
For petlja-interval	<pre>for i in range (0,10):</pre>
	<pre>lokalno_vrijeme = time.asctime()</pre>
Dohvaćanje vremena	<pre>print("Podsjetnik-uzeti napitak")</pre>
Vrijeme-ograničenje	<pre>print(lokalno_vrijeme)</pre>
Ispis vremena i poruke	time.sleep(a)

Zadatak 0-Uvodni\_motivacijski

Nakon što sam s učenicima prokomentirao zadatak, zadajem im da zadatak prepišu i testiraju na svojim računalima.

N: Prepišite zadatak i omogućite ispis poruke svake 4 sekunde.

Dok učenici rade, obilazim ih i pratim njihov rad. Po potrebi interveniram.

Nakon uvodnog (motivacijskog) zadatka, prelazim na rješavanje zadataka s for petljom. Tekst zadatka učenicima prikazujem na ppt prezentaciji. Prije rješavanja svakog zadatka na ploči s učenicima prokomentiram zadatak te zajednički napravim nacrt rješenja.

Nakon toga učenici sami rješavaju zadatak. Za vrijeme rješavanja, prolazim razredom, pratim rad učenika, po potrebi interveniram. Ukoliko zamijetim da više učenika na nekom mjestu radi pogrešku ili ima poteškoće u radu, privremeno zaustavljam rad i javno sa svim učenicima razmotrim tu pogrešku ili problem.

Zadatak 1. (trajanje do 6 minuta) Napisati program koji će omogućiti unos prirodnog broja n i ispisati djelitelje tog broja n. *referenciram se na tekst zadatka Primjeri.pptx slajd 2.* 

Zajedno s učenicima radim nacrt rješenja na ploči.



Nacrt rješenja na ploči

N: Pronaći djelitelje nekog jednoznamenkastog i dvoznamenkasto prirodnog broja je dosta jednostavno i možemo to obaviti vrlo brzo. No da se od nas traži pronaći sve djelitelje nekog troznamenkastog broja ili višeznamenkastog broja; to bi potrajalo i bilo dosta naporno. Zamislimo da treba pronaći sve djelitelje broja 347. Neke djelitelje bi odmah izračunali no poneke bi ručno raspisali.

N: Da si olakšamo posao, napisati ćemo program koji će pronaći sve djelitelje nekog zadanog prirodnog broja n.

#### N: Kada neki broj postaje djelitelj nekog prirodnog broja n?

U: Ako je ostatak dijeljenja prirodnog broja n i djelitelja jednak nuli.

N: Izvrsno. Dakle svi brojevi kojima je nakon dijeljenja prirodnog broja sa skupom brojeva do tog broja n ostatak jedan nuli, nazivamo djeliteljima.

N: Govorimo o dijeljenju s ostatkom.

#### N: Koja aritmetička operacija omogućava dijeljenje s ostatkom?

U: Operacija modulo omogućava dijeljenje s ostatkom.

#### N: Možemo li pomoću parametara for petlje provjeriti djelitelje nekog prirodnog broja?

U: Ne možemo provjeriti djelitelje nekog prirodnog broja pomoću for petlje.

#### N: Pomoću koje programske strukture bi napravili provjeru?

U: Provjeru bi napravili pomoću grananja.

N: Tako je. Odluke će nam poslužiti za ispitivanje ostatka dijeljenja prirodnog broja n sa skupom brojeva do n.

#### N: Kako bi onda glasio uvjet grananja?

U: Uvjet grananja bi glasio: neki n % brojem iz skupa brojeva jednak je nuli.

N: Izvrsno. To bi bio uvjet.

N: Dakle djelitelje dobivamo na način da prođemo cijelim skupom brojeva od 1 do n i za svaki provjerimo ostatak dijeljenja. Ako nema ostatka, broj je sigurno djelitelj prirodnog broja n.

N: Budući da prolazimo cijelim skupom brojeva, potrebna nam je petlja.

#### N: Koje ćemo sve parametre petlje koristiti?

U: Koristiti ćemo početnu i završnu vrijednost kao parametre petlje.

N: Točno. Dakle trebamo ograničiti interval petlje na željeni skup brojeva.

N: Korisniku dajemo mogućnost unosa prirodnog broja. Što trebamo učiniti prije određivanja završne vrijednosti petlje?

U: Trebamo omogućiti naredbom upisa unos željene vrijednosti.

N: Izvrsno. Znači u završnoj vrijednosti petlje bi pisalo na primjer n.

#### N: Ako bi u for petlji interval postavili između 1 i n, hoće li završna vrijednost ući u brojanje?

U: Završna vrijednost ne bi ušla u brojanje.

**N:** Tako je. Ako bi krenuli brojati od nule završna vrijednost ne bi ušla u brojanje to jest završna vrijednost bi bila n-1.

#### N:Što trebamo učiniti da bi broj n kao završna vrijednost ušao u brojanje?

U: Potrebno je završnu vrijednost uvećati za 1.

N: Izvrsno. Završna vrijednost petlje treba biti uvećana za 1. Tek onda će u brojanje ući i broj n.

N: I naravno na kraju trebamo ispisati sve djeljitelje broja n.

#### N: Imate li još kakvih pitanja?

Ukoliko pitanja postoje odgovaram na njih, u suprotnom dajem uputu za rješavanje zadatka.

Kao pomoć u rješavanju otvaram slajd s elementarnim koracima za rješavanje zadatka Primjeri.pptx slajd 3. Za vrijeme rješavanja, pratim rad učenika, po potrebi individualno interveniram. Ukoliko zamijetim pogreške kod više učenika, zaustavljam na trenutak rad i javno sa svim učenicima razmotrim tu pogrešku ili problem.

```
n = int(input("Unesi prirodan broj"))
for i in range(1,n+1):
    if n%i==0:
        print(i, end=' ')
```

#### Zadatak 1- rješenje

Po isteku 7 minuta predviđenih za rješavanje zadatka demonstriram rješenje na nastavničkom računalu preko online vizualizatora na poveznici <u>https://goo.gl/T0ZqS1</u> i komentiram rješenje. Ukoliko ima potrebe, ističem neka kreativna rješenja učenika koja pohvaljujem.

Zadatak 2.(trajanje rješavanja do 6 minuta) Nastavnik ispituje učenike na satu. Ispitati će svakog ntog učenika u imeniku. Napisati program koji će omogućiti unos broja učenika te prirodan broj n kao kriterij odabira učenika iz imenika. Program treba ispisivati redne brojeve učenika koje će nastavnik ispitati. *referenciram se na tekst zadatka Primjeri.pptx slajd 4.* 

Zajedno s učenicima radim nacrt rješenja na ploči.



Nacrt rješenja na ploči

N: Svatko od vas u imeniku ima svoj redni broj. Nastavnik usmeno ispituje učenike, ali prije nego što nekog prozove pita da li se netko javlja za usmeno odgovaranje.

N: Ukoliko nema dobrovoljaca, nastavnik će sam odabrati učenike za usmeno odgovaranje. Primjerice od 20 učenika, nastavnik želi ispitati svakog šestog učenika prema rednom broju u imeniku. Naravno nastavnik se može predomisliti i odlučiti ispitati svakog 4 učenika po rednom broju imenika.

N: Idemo kreirati rješenje.

# N: Prema tekstu zadatka, prije nego odredimo učenike koje će nastavnik ispitati, što trebamo napraviti?

U: Trebamo unijeti broj učenika.

N: Tako je. Prvo moramo odrediti ukupan broj učenika nad kojima će se napraviti izbor kandidata za usmeno ispitivanje.

#### N: Gdje ćemo u for petlji ograničiti ukupan broj učenika?

U: U intervalu range.

N: Odgovor je točan, ali bi trebao biti precizniji.

#### N: Na koje mjesto unutar range intervala unosimo broj učenika?

U: Na mjesto završne vrijednosti.

N: Izvrsno. Na mjesto završne vrijednosti ćemo unijeti koliko učenika u grupi ukupno ima. Znači to je ukupan broj učenika koji unosi nastavnik.

#### N: Koliko vas sada ima u grupi?

U: U grupi nas je 14.

N: Dakle zadali bi završnu vrijednost na primjer na ukupno 14 učenika.

N: Uvažavajući samo parametre petlje kako ćemo odrediti da se ispisuje svaki 6 učenik po rednom broju u imeniku?

U: Trebamo odrediti korak petlje.

#### N: Dobro. Što bi to značilo konkretno u for petlji?

U: To bi značilo da na mjestu koraka petlje upisujemo broj 6.

N: Da. To bi generalno bilo točno. No prema zadatku, nastavniku dajemo mogućnost da sam bira korak petlje. Dakle, treba unijeti prirodan broj **n** kao **kriterij** odabira učenika iz imenika.

N: Znači nastavnik se može predomisliti i reći: ispitivati ću svakog 2. učenika po rednom broju imenika.

N: Upravo to znači onaj prirodan broj n u zadatku. Kako ćemo u kodu omogućiti određivanje koraka petlje na taj način?

U: Napisati ćemo naredbu za unos koraka petlje.

N: Izvrsno. Znači preko tipkovnice ćemo unijeti željeni korak petlje odnosno željeni prirodan broj prema kojem će se ispisivati redni brojevi učenika iz imenika.

# N: Što ćemo uvrstiti u naredbu ispisa da nam se ispišu redni brojevi učenika? Hoće li to biti jedan od parametara petlje?

U: U ispisu ćemo uvrstiti brojač petlje.

N: Izvrsno. Brojač broji svaki korak petlje odnosno svaku vrijednost zapisanu na poziciji određenoj korakom petlje. Tako ćemo dobiti ispis rednih brojeva učenika koje će nastavnik ispitati.

N: Ne zaboravimo da u brojanje ulazi i završna vrijednost petlje.

N: Prema tome, kako trebamo napisati izraz završne vrijednosti petlje?

U: Završna vrijednost petlje bi glasila: završna vrijednost+1.

N: Tako je. Završnu vrijednost trebamo uvećati za 1 kako bi u brojanje ušla i završna vrijednost.

#### N: Imate li možda još kakvih pitanja?

Ukoliko pitanja postoje odgovaram na njih, u suprotnom dajem uputu za rješavanje zadatka.

Kao pomoć u rješavanju otvaram slajd s elementarnim koracima za rješavanje zadatka Primjeri.pptx slajd 5. Za vrijeme rješavanja, pratim rad učenika, po potrebi individualno interveniram. Ukoliko zamijetim pogreške kod više učenika, zaustavljam na trenutak rad i javno sa svim učenicima razmotrim tu pogrešku ili problem.

```
uc=int(input("Unesite broj učenika"))
n=int(input("Unesite prirodan broj"))
for i in range(1,uc+1, n):
print("Ispitivati će se učenici s rednim brojevima:", i)
```

#### Zadatak 2- rješenje

Po isteku šest minuta predviđene za rješavanje zadatka demonstriram rješenje na nastavničkom računalu preko online vizualizatora na poveznici <u>https://goo.gl/L9Yiw8</u> i komentiram rješenje. Ukoliko ima potrebe, ističem neka kreativna rješenja učenika koja pohvaljujem.

Zadatak 3.(trajanje rješavanja do 7 minuta) Napisati program koji simulira bacanje novčića n puta te ispisati koliko puta se pojavljuje pismo i glave. *referenciram se na tekst zadatka Primjeri.pptx slajd 6.* 

Zajedno s učenicima radim nacrt rješenja na ploči.



Nacrt rješenja na ploči

N: Ideja zadatka je da ispitamo koja je vjerojatnost da se kod određenog broja bacanja n pojavi pismo ili glava.

N: Bacanje pisma odnosno glave se odvija nasumično. Na primjer može nekoliko puta pasti samo glava ili nasumična kombinacija glave i pisma, a opet može padati samo glava.

#### N: Što prvo trebamo omogućiti?

U: Prvo trebamo omogućiti unos bacanja.

N: Tako je. Prvo u program trebamo unijeti određeni broj bacanja.

#### N: Što bismo u parametre for petlje upisali?

U: Upisali bismo neki broj bacanja.

N: Tako je. Dakle ne trebamo pisati intervale. Dovoljno je napisati samo ukupan broj bacanja to jest odrediti završnu vrijednost. Trenutno vrijednost brojača koristi samo za brojanje bacanja i ni za što drugo.

N: Novčić ima samo dvije strane: pismo i glavu, odnosno može imati dva stanja.

# N: Budući da imamo dva stanja, s kojim vrijednostima bismo u programiranju predočili pismo odnosno glavu?

U: Pismo i glavu možemo predočiti kao vrijednosti 0 i 1.

N: Izvrsno. Na primjer 0 može biti pismo, a 1 glava.

# N: Pojave glave i pisma su nasumične. Za rad s nasumičnim brojevima, koju funkciju ćemo koristiti?

U: Koristiti ćemo funkciju randint.

### N: Dobro. Da bismo koristili funkciju randint, što trebamo učiniti?

U: Trebamo uključiti biblioteku za rad s nasumičnim brojevima.

#### N: Kako se biblioteka zove?

U: Biblioteka se zove random.

#### N:Tako je. Koje ćemo vrijednosti unijeti u interval randint?

- U: U interval ćemo unijeti 0 i 1.
- N: Tako je. Dakle dva stanja simuliraju dvije strane novčića: pisma i glave.

N: U sljedećem koraku bi morali pobrojiti koliko se puta pojavi 0 ili 1 odnosno pismo ili glava.

#### N: Gdje ćemo zapisati koliko se puta pojavljuje pismo, a koliko glava?

U: Zapisati ćemo u varijable.

N: Tako je. Varijable će poslužiti kao brojači.

#### N: Koliko je varijabli potrebno?

U: Potrebne su dvije varijable.

N: Izvrsno. U jednu ćemo zapisati broj pojave pisma, a u drugu broj pojave glave.

N: Varijablu namijenjenu za spremanje sadržaja u početku ćemo postaviti na nulu.

#### N: Što bi se dogodilo da kod inicijalizacije varijablu postavimo na vrijednost 1?

U: To znači da bi se prva vrijednost koju pribrajamo toj varijabli uvećala za jedan.

N: Da. Kada je početni sadržaj varijable jednak 1, onda bi se u toj varijabli zapisana prva vrijednost početnog kruga petlje uvećala za taj jedan. Kada je početni sadržaj varijable jednak 0, onda se u tu varijablu zapisuje prva vrijednost u početnom krugu petlje bez ikakvih uvećanja zato jer je nula neutralni broj.

N: Sadržaj jedne varijable za spremanje pisma uvećati ćemo za jedan kada randint generira vrijednost 0, a sadržaj varijable za spremanje glave kada randint generira vrijednost 1.

N: Drugim riječima sadržaje varijabli uvećavamo za jedan ovisno o generatoru nasumičnih brojeva.

# N: Kako bi onda izgledao izraz za pribrajanje vrijednosti pisma odnosno glave u programu? Možemo kao varijablu uzeti x.

U: Izraz bi izgledao x=x+1

#### N: Tako je. Kako bi to objasnili?

U: Vrijednost varijable se uvećava za 1 kada se pojavi pismo odnosno glava.

N: Točno, samo podsjećam da imamo dvije varijable. Kada se pojavi pismo odnosno vrijednost 0, postojeća vrijednost varijable se uveća za 1; kada se pojavi glava odnosno vrijednost 1, postojeća vrijednost druge varijable se uveća za 1. U prvom slučaju se x uveća za jedan, u nekom drugom krugu petlje na primjer y se uveća za jedan.

#### N: Kako bi izračunali vjerojatnost pojave pisma odnosno glave?

U: Vjerojatnost bi izračunali kao broj pisma odnosno glave podijeljeno s brojem bacanja.

N: Izvrsno. Znači formula je ovakva: ukupan broj pojave pisma ili glave / broj bacanja.

#### N: Imate li još kakvih pitanja?

Ukoliko pitanja postoje odgovaram na njih, u suprotnom dajem uputu za rješavanje zadatka.

Kao pomoć u rješavanju otvaram slajd s elementarnim koracima za rješavanje zadatka Primjeri.pptx slajd 7. Za vrijeme rješavanja, pratim rad učenika, po potrebi individualno interveniram. Ukoliko zamijetim pogreške kod više učenika, zaustavljam na trenutak rad i javno sa svim učenicima razmotrim tu pogrešku ili problem.



Zadatak 3- rješenje

Po isteku 7 minuta predviđenih za rješavanje zadatka demonstriram rješenje na nastavničkom računalu preko online vizualizatora na poveznici <u>https://goo.gl/buiehX</u> i komentiram rješenje. Ukoliko ima potrebe, ističem neka kreativna rješenja učenika koja pohvaljujem.

N: U prethodna dva zadatka, radili smo pretežito s brojevima. Uz pomoću for petlje moguće je ponavljati i ispisivanje odgovarajućih poruka kao što smo vidjeli u uvodnom primjeru. Sljedeći zadatak upravo obrađuje tu problematiku.

Zadatak 4.(trajanje rješavanja do 6 minuta) Napišite program koji će unositi broj učenika te omogućiti unos visine učenika prema broju unesenih učenika. Program treba ispisati visinu najnižeg i najvišeg učenika.

referenciram se na tekst zadatka Primjeri.pptx slajd 8. Zajedno s učenicima radim nacrt rješenja na ploči.



Nacrt rješenja na ploči

N: Primijetimo da u ovom zadatku treba napraviti nekoliko ključnih stvari: omogućiti unos broja učenika, omogućiti unos visina, razvrstati visine i ispisati visinu najvišeg i najnižeg učenika.

# N: Što trebamo učiniti u početku?

U: U početku trebamo unijeti broj učenika.

N: Tako je. **Gdje trebamo označiti u kodu da unosimo visinu učenika prema određenom broju učenika**?

U: Trebamo završnu vrijednost petlje postaviti na broj učenika.

# N: Možemo li provjeru najvišeg i najnižeg učenika napraviti u samoj for petlji u nekom od parametara?

U: Ne možemo napraviti provjeru pomoću parametara for petlje.

N: Točno.

### N: Kako ćemo onda provjeriti tko je najviši, a tko najniži učenik?

U: Najvišeg i najnižeg učenika možemo provjeriti pomoću odluka.

N: Naravno. Provjeru možemo napraviti pomoću odluka.

# N: Kako ćemo provjeriti unesene vrijednosti da dođemo do zaključka koja je najviša, a koja najniža visina?

U: Trebamo ih međusobno uspoređivati.

# N: Izvrsno. Treba ih međusobno usporediti. Kako će izgledati usporedba na primjer za najnižeg učenika?

U: Svaku unesenu vrijednost treba usporediti s trenutnom vrijednosti varijable u koju je spremljena visina.

N: Dobro. To je točno, a ujedno i jedan dio odgovora.

N: Isto trebamo učiniti i za provjeru najvišeg učenika. Trebamo obje varijable: za najnižeg i najvišeg učenika, popuniti sadržajem prvog unosa visine da bi mogli krenuti s prvim krugom usporedbe.

#### N: Prema tome koliko bi nam trebalo if uvjetovanja da provjerimo visine učenika?

U: Potrebna su dva uvjetovanja.

N: Dobro. U jednom uvjetovanju bi provjeravali najnižu, a u drugom najvišu vrijednost. Tu su nam potrebna dva uvjetovanja.

N: Maloprije smo govorili o tome da varijable moraju biti popunjene sadržajem radi usporedbe visina u idućem krugu petlje.

### N: Kako ćemo popuniti te dvije varijable?

U: Kada korisnik unese prvu vrijednost onda se ona sprema u obje varijable.

### N: U redu je. To je sasvim točan odgovor. Kada korisnik unosi prvu vrijednost varijabli?

U: Korisnik unosi prvu vrijednost varijabli u prvom krugu petlje.

N: Tako je. Dakle prvi krug petlje je ključan za zapisivanje punjenje varijabli visina.

#### N: Gdje je zapisana vrijednost prvog kruga petlje?

U: Vrijednost je zapisana u brojaču petlje.

#### N: Izvrsno. Prema tome što moramo učiniti?

U: moramo postaviti uvjet za punjenje varijabli s početnom visinom učenika.

#### N:Kako bi izgledao uvjet za ostvarenje toga?

U: Brojač petlje bi trebao biti jednak početnoj vrijednosti.

N: Izvrsno. Dakle u trenutku kada se brojač petlje nalazi u početnoj vrijednosti petlje, možemo obje varijable popuniti prvom unesenom visinom.

#### N: Koliko ćemo onda uvjetovanja koristiti?

U: Koristiti ćemo ukupno tri uvjetovanja.

#### N: Imate li možda još kakvih pitanja prije rješavanja zadatka?

Ukoliko pitanja postoje odgovaram na njih, u suprotnom dajem uputu za rješavanje zadatka.

Kao pomoć u rješavanju otvaram slajd s elementarnim koracima za rješavanje zadatka Primjeri.pptx slajd 9. Za vrijeme rješavanja, pratim rad učenika, po potrebi individualno interveniram. Ukoliko zamijetim pogreške kod više učenika, zaustavljam na trenutak rad i javno sa svim učenicima razmotrim tu pogrešku ili problem.



Zadatak 4- rješenje

Po isteku 6 minuta predviđenih za rješavanje zadatka demonstriram rješenje na nastavničkom računalu preko online vizualizatora na poveznici <u>https://goo.gl/hd8MOI</u> i komentiram rješenje. Ukoliko ima potrebe, ističem neka kreativna rješenja učenika koja pohvaljujem.

N: Prethodni primjer nadograđujemo s malo više zahtjeva.

Zadatak 5.(trajanje rješavanja do 8 minuta) Napisati program za učitavanje temperature zraka u prethodnih 10 dana te ispisati srednju temperaturu iznad nule i srednju temperaturu ispod nule.

referenciram se na tekst zadatka Primjeri.pptx slajd 10. Zajedno s učenicima radim nacrt rješenja na ploči.



Nacrt rješenja na ploči

N: U sljedećem zadatku trebamo izračunati srednje vrijednosti temperatura iznad i ispod nule.

#### N: Kako glasi sam algoritam za računanje srednje vrijednosti?

U: Algoritam za računanje srednje vrijednosti glasi: zbroj brojeva podijeljen s brojem tih brojeva.

N: Izvrsno. Dakle algoritam za izračunavanje srednje vrijednosti možemo predočiti kao: zbroj brojeva/ukupan broj brojeva.

#### N: Na koji način ćemo razvrstati unesene vrijednosti na veće od nule i manje od nule?

U: Unose brojeva ćemo razvrstati pomoću odluka.

#### N: Izvrsno. Koliko uvjeta moramo postaviti?

U: Moramo postaviti dva uvjeta.

N: Točan odgovor. Moramo postaviti dva uvjeta.

N: Razmislimo prema tekstu zadatka, koliko ćemo varijabli koristiti za pohranu vrijednosti kod računanja srednje vrijednosti temperature iznad nule? Prisjetimo se algoritma za izračunavanje srednje vrijednosti.

U: Za pohranu vrijednosti kod računanja srednje vrijednosti, koristiti ćemo dvije varijable.

N: Izvrsno. Dakle koristiti ćemo dvije varijable za računanje jedne srednje vrijednosti.

#### N: Što ćemo u koju varijablu spremiti?

U: U jednu varijablu ćemo spremiti zbroj vrijednosti, a u drugu ukupan broj unesenih vrijednosti.

N: Izvrsno.

#### N: Prisjetimo se zadatka 2. U kakvom obliku smo uvećavali sadržaj varijable za 1?

- U: Sadržaj smo spremali u obliku x=x+1.
- N: Dobro. Znači postojeći sadržaj varijable x uvećali smo za jedan.
- N: Dakle takav način uvećanja sadržaja varijable smo koristili za pobrojavanje.

#### N: Možemo li isti oblik koristiti i za zbrajanje unesenih vrijednosti?

U:Ne možemo.

#### N: Točno. Što trebamo učiniti da bismo zbrojili unesene vrijednosti nakon ispitivanja uvjeta?

U: Potrebno je vrijednosti varijable za zbrajanje pridružiti trenutačnu unesenu vrijednost.

#### N: Tako je. Kako bi glasio onda izraz za zbrajanje?

U: Izraz za zbrajanje bi glasio x=x+unos.

#### N: Izvrsno.

N: Sada kada smo razvrstali brojeve i spremili ih u pripadajuće varijable, moramo još izračunati srednje vrijednosti prema algoritmu i napraviti ispis.

N: Osim toga programsku petlju ćemo ograničiti na 10 unosa.

### N: Imate li možda još kakvih pitanja prije rješavanja zadatka?

Ukoliko pitanja postoje odgovaram na njih, u suprotnom dajem uputu za rješavanje zadatka.

Kao pomoć u rješavanju otvaram slajd s elementarnim koracima za rješavanje zadatka Primjeri.pptx slajd 11. Za vrijeme rješavanja, pratim rad učenika, po potrebi individualno interveniram. Ukoliko zamijetim pogreške kod više učenika, zaustavljam na trenutak rad i javno sa svim učenicima razmotrim tu pogrešku ili problem.

brp=0 sp=0 brn=0 sn=0
<pre>for i in range (0,10): unos=int(input("Unesi broj ")) if unos &gt; 0: sp=sp+unos brp=brp+1 if unos &lt; 0: sn=sn+unos brn=brn+1</pre>
arp=sp/brp arn=sn/brn
print(" Srednja temperatura iznad nule ", arp) print(" Srednja temperatura ispod nule ", arn)

Zadatak 5- rješenje

Po isteku 8 minuta predviđenih za rješavanje zadatka demonstriram rješenje na nastavničkom računalu preko online vizualizatora na poveznici <u>https://goo.gl/qll7Dg</u> i komentiram rješenje. Ukoliko ima potrebe, ističem neka kreativna rješenja učenika koja pohvaljujem.

Kao rezervu, imam spreman zadatak broj 6 (referenca Primjeri.pptx slajd 12). Taj zadatak je opcionalan.Ovisno o uspješnosti odrađivanja ostalih zadataka, odlučujem o rješavanju zadatka 6. Dakle ukoliko ostane vremena, s učenicima rješavam zadatak. Predviđeno vrijeme trajanja rješavanja zadatka je 5 minuta.

Zadatak 6 (trajanje do 5 minuta) Napisati program koji će generirati tablicu množenja brojeva do 10.

referenciram se na tekst zadatka Primjeri.pptx slajd 12. Zajedno s učenicima radim nacrt rješenja na ploči.



#### Nacrt rješenja na ploči

N: U zadatku se traži generirati tablicu množenja do broja 10. Tablica s vrijednostima množenja treba izgledati poput tablice na prezentaciji. *Referenciram se na prezentaciju Primjeri.pptx slajd 12.* 

#### N: Od čega se sastoje tablice?

U: Tablice se sastoje do redaka i stupaca.

- N: Tako je. Svaka tablica ima i redak i stupac.
- N: U tablici na prezentaciji, retci i stupci su istaknuti crvenim okvirima.

#### N: Kako je dobivena vrijednost 30?

U: Vrijednost trideset je dobivena množenjem 5\*6.

#### N: Tako je. Što predstavlja 5\*6?

U:Pet predstavlja peti redak, a šest šesti stupac.

N: Izvrsno. Znači da bi dobili vrijednost 30, pomnožili smo vrijednost 5. retka sa vrijednošću 6. stupca.

N: Ideja zadatka je upravo generirati tablicu množenja na temelju množenja vrijednosti redaka i stupaca.

#### N: Budući da govorimo o retku i stupcu, koliko for petlji tebamo za generiranje retka i stupca?

U: Trebamo dvije for petlje.

N: Izvrsno. Pomoću jedne petlje generiramo retke, a pomoću druge stupce.

N:Govorimo o množenju brojeva do 10. Na koji interval trebamo ograničiti petlje?

U: Petlje trebamo ograničiti na interval 10.

N: Kao početnu vrijednost ćemo uzeti 1 jer s njime započinjemo množenje.

# N:Ako je početna vrijednost 1, a množimo brojeve do 10, hoće li završna vrijednost biti postavljena na 10?

U: Završna vrijednost neće biti postavljena na 10.

#### N: Tako je. Zašto ne smijemo završnu vrijednost postaviti na 10?

U: Završnu vrijednost ne smijemo postaviti na 10 zato jer ta vrijednost neće ući u brojanje petlje.

N: Dobro. Prema tome završnu vrijednost trebamo odrediti kao izraz n+1.

#### N: Što taj n+1 u našem primjeru znači?

U: n+1 u našem primjeru znači da završnu vrijednost postavljamo na 11.

N: Dobro. Taj dio smo zasad riješili.

N: Vrijednost tablice dobivamo kao umnožak vrijednosti retka sa vrijednošću stupca.

# N: Koju varijablu for petlje trebamo uzeti u obzir kako bismo mogli brojati vrijednosti stupaca i redaka?

U: Trebamo uzeti u obzir brojač petlje i.

N: Točno. U brojaču i je spremljen trenutni broj koraka petlje.

N: Radi lakšeg snalaženja u kodu, brojače petlji ćemo nazvati različitim imenima: red i stupac.

N: For petlje trebamo međusobno uskladiti.Na primjer generiramo vrijednosti za prvi redak i prvi stupac.

#### N: Koja je početna vrijednost prvog retka prema tablici iz prezentacije?

U: Početna vrijednost je 1.

#### N: Tako je. Što je sada potrebno učiniti da bismo dobili u tom istom retku ostale vrijednosti retka?

U: Potrebno je početnu vrijednost pomnožiti s intervalom od 10 brojeva.

#### N: Izvrsno. Gdje je taj interval određen?

U: Taj interval je određen u drugoj petlji.

N: Tako je. Ta druga petlja je petlja stupca.

N: Primijetimo da smo vrijednost retka uzeli kao neku "konstantu" tog retka koju smo množili sa svim brojevima stupaca u intervalu od 10 brojeva.

N: Stoga imamo dvije petlje: petlju za redak i stupac.

N: Petlja za redak u brojaču sadržava vrijednost koja se množi sa svakom vrijednosti brojača petlje za stupac u intervalu od 10 brojeva. Tek kada petlja za stupac izvrši naredbu množenja retka sa stupcem i dođe do završne vrijednosti, brojač petlje za redak se uveća za jedan odnosno prelazi u idući redak.

#### N: Prema tome, gdje se petlja za stupac mora u kodu nalaziti?

U: Petlja za stupac se mora onda nalaziti u tijelu for petlje za redak.

N:Izvrsno. Dakle petlju za stupac bi smjestili u tijelo petlje za redak.

#### N: U tijelu koje petlje bi se izvršavala operacija množenja retka sa stupcem?

U: Operacija množenja bi se izvršavala u petlji za stupac.

#### N: Tako je. Što bi se dogodilo kada bi u for petlji za redak brojač bio postavljen na vrijednost 3?

U: Svaka vrijednost brojača petlje za stupac bi se množila s vrijednosti 3 brojača petlje za redak.

N: Izvrsno. Upravo smo kreirali algoritam za izradu tablice množenja.

#### N: Imate li možda još kakvih pitanja prije rješavanja zadatka?

Ukoliko pitanja postoje odgovaram na njih, u suprotnom dajem uputu za rješavanje zadatka.

Kao pomoć u rješavanju otvaram slajd s elementarnim koracima za rješavanje zadatka Primjeri.pptx slajd 13. Za vrijeme rješavanja, pratim rad učenika, po potrebi individualno interveniram. Ukoliko zamijetim pogreške kod više učenika, zaustavljam na trenutak rad i javno sa svim učenicima razmotrim tu pogrešku ili problem.

#### Zadatak 6- rješenje

Po isteku 5 minuta predviđenih za rješavanje zadatka demonstriram rješenje na nastavničkom računalu preko online vizualizatora na poveznici <u>https://goo.gl/qll7Dg</u> i komentiram rješenje. Ukoliko ima potrebe, ističem neka kreativna rješenja učenika koja pohvaljujem.

# ZAVRŠNI DIO (trajanje do 4 minute)

U završnom dijelu zamolim dvoje najbližih učenika da podijele nastavne listiće sa zadacima. Referenca dokument (NASTAVNI\_LISTIĆ\_Domaća\_zadaća). Domaća zadaća se sastoji od tri zadatka. Prva dva zadatka se temelje na razumijevanju, predočavanju apstrakcije te kreativnoj konkretizaciji rješenja. Treći zadatak podrazumijeva provjeru razumijevanja programskog koda, poznavanje sintakse i detekciju pogrešaka. Prije kraja sata dajem kratke upute za rješavanje zadataka učenicima.

N: Za domaću zadaću imate tri zadatka. Prva dva zadatka su vrlo slična zadacima koje smo danas rješavali.

N:U trećem zadatku se nalaze pogreška u programskom kodu. Potrebno je poznavati sintaksu i razumjeti što program radi.

N: Ukoliko ne možete otkriti grešku, preporučam da program prepišete u Pythonu i testirate ga.

#### N: Imate li kakvih pitanja?

Ukoliko ima pitanja, odgovaram na njih; u suprotnom se zahvaljujem učenicima na suradnji, dajem uputu o pospremanju radnog mjesta i isključivanja računala.

N: Hvala vam na suradnji i odrađenom satu. Molim pospremite radna mjesta i isključite računala.

N: Doviđenja.

U: Doviđenja.
# PLAN PLOČE



254

# PRILOZI

### Uvodni

```
import time
a=2
for i in range (0,10):
    lokalno_vrijeme = time.asctime()
    print("Podsjetnik-uzeti napitak")
    print(lokalno_vrijeme)
    time.sleep(a)
```

Zadatak	1
---------	---

```
n = int(input("Unesi prirodan broj"))
for i in range(1,n+1):
    if n%i==0:
        print(i, end=' ')
```

#### Zadatak 2

```
uc=int(input("Unesite broj učenika"))
n=int(input("Unesite prirodan broj"))
for i in range(1,uc+1, n):
    print("Ispitivati će se učenici s rednim brojevima:", i)
```

#### Zadatak 3

```
from random import randint
n=int(input("Unesite željeni broj "))
br_pismo=0
br_glava=0
for i in range(n):
    bacanje=randint(0,1)
    if bacanje == 0:
        br_pismo = br_pismo + 1
    else:
        br_glava = br_glava + 1
vjer_pismo=br_pismo/n
vjer_glava=br_glava/n
print ("Vjerojatnost pismo", vjer_pismo)
print ("Vjerojatnost glava", vjer_glava)
```

Zadatak 4

```
najnizi=0
najvisi=0
n=int(input("Koliko učenika želite unijeti: "))
for i in range(0,n):
    u=int(input("Unesite visinu ucenika: "))
    if i==0:
        najnizi=najvisi=u
    if najnizi>u:
        najnizi=u
    if najvisi<u:
        najvisi=u
print ("Najniži:", najnizi, "cm")
print ("Najviši:", najvisi, "cm")
```

Zadatak 5

```
brp=0
sp=0
brn=0
sn=0
for i in range (0,10):
    unos=int(input("Unesi broj "))
    if unos > 0:
        sp=sp+unos
        brp=brp+1
    if unos < 0:
        sn=sn+unos
        brn=brn+1
arp=sp/brp
arn=sn/brn
print(" Srednja temperatura iznad nule ", arp)
print(" Srednja temperatura ispod nule ", arn)
```

Zadatak 6-rezerva

```
for red in range(1,11):
    for stupac in range(1,11):
        print("{0:5}".format(red*stupac), end="")
    print()
```

#### O autoru:



Zoran Hercigonja, rođen je 13.04.1990. godine u Varaždinu. Osnovnu školu "Petar Zrinski " završio je 2005. godine u Jalžabetu. Iste godine upisuje Elektrostrojarsku školu Varaždin, smjer "Tehničar za računalstvo" i uspješno položenom maturom završava 2009. godine. Daljnje školovanje nastavlja na Fakultetu organizacije i informatike Varaždin 2009. godine.Preddiplomski studij

"Informacijski sustavi" završava 2013. godine. Nastavlja studirati na istom fakultetu na diplomskom studiju za informatičare usmjerenja "Informatika u obrazovanju" koji završava 2015. godine s velikom pohvalom (Magna cum laude). Akademske godine 2013/2014. dobitnik je dekanove nagrade za najboljeg studenta diplomskog studija informatike. Pripravnički staž obavlja na Drugoj gimnaziji Varaždin u razdoblju između 2015. i 2016. godine. Državno-stručni ispit za nastavnika informatike uspješno polaže na V. gimnaziji Zagreb 2017. godine. Radi kao profesor informatike i matematike. Autor je više stručnih radova koje redovito objavljuje u časopisima: Pogled kroz prozor, International Journal of Digital Technology and Economy, Matematika i škola, zborniku Carnetove korisničke konferencije te u Zborniku radova Veleučilišta u Šibeniku. Također je jedan od nagrađenih sudionika u izradi metodičke preporuke iz nastave informatike u sklopu kurikularne reforme "Škola za život".

## ISBN 978-953-8200-15-1