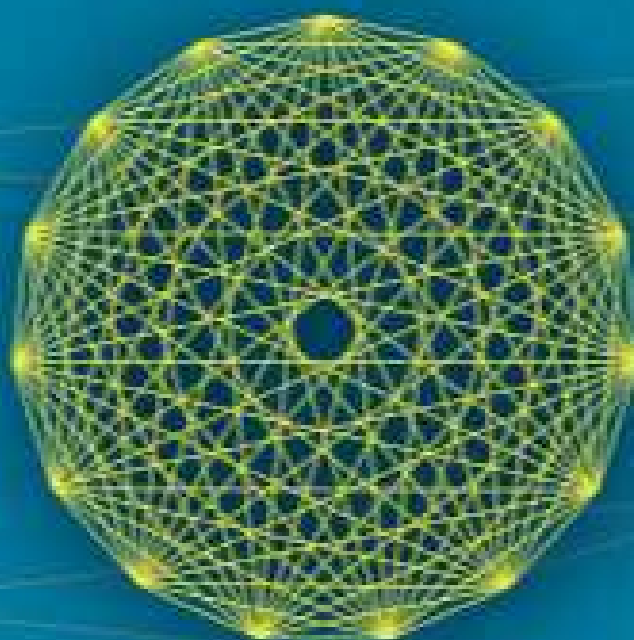


LIJAN 2014.

CYBERNETICA

BILTEN DRUŠTVA KIBERNETIČARA - RIJEKA



MARINA ČIČIĆ-SAIN
SNJEŽANA BABIĆ:

POČETNICA - MEMA

PRIRUČNIK ZA UČITELJICE I UČITELJE

Izdavač: Društvo kibernetičara
Broj primjeraka: 100
Tisak: Futura d.o.o. Rijeka

Design ovitka: Jelena Babić dipl. diz.
Publikacija se izdaje uz financijsku potporu Primorsko-goranske županije i grada Rijeke

CIP zapis dostupan u računalnom katalogu Sveučilišne knjižnice Rijeka pod brojem 130430060.

ISBN 978-953-95663-4-8

Copyright © Marina Čičin-Šain

Ova je knjiga intelektualno vlasništvo. Niti jedan dio ove knjige ne smije se reproducirati u bilo kojem obliku ili na bilo koji način, niti pohranjivati u bazu podataka bilo koje namjene bez prethodnog pismenog dopuštenja autorice, osim u slučajevima kratkih navoda. Izrada kopija bilo kojeg dijela knjige je zabranjena.

Počelnica programiranja

Marina Čičin-Šain
Marina.cicinsain@gmail.com
Snježana Babić
sbabicv@gmail.com

Sadržaj

Uvod	58
Opis metode	60
1. Gradivo prvog razreda	62
1.1 Nastavna tema: Uvod	62
1.2 Nastavna tema: Izrada Meme	63
1.3 Nastavna tema: Ulaz	65
1.4 Nastavna tema: Izlaz	69
1.5 Nastavna tema: Obrada	72
1.6 Nastavna tema: Igrkaz "Moje računalo"	76
1.7 Nastavna tema: Napredniji rad s MEMOM	81
1.8 Nastavna tema: Sljedbenik i prethodnik	84
1.9 Nastavna tema: Ponavljanje gradiva	87
1.10 Nastavna tema: Provjera znanja	89
2. Gradivo drugog razreda	99
2.1 Nastavna tema: Ponavljanje	99
2.2 Nastavna tema: Adresiranje pretinaca tekstem ...	100
2.3 Nastavna tema: Kopiranje sadržaja pretinca	101
2.4 Nastavna tema: Mema uz tablicu množenja	104
2.5 Nastavna tema: Logički operatori	110
3. Gradivo trećeg razreda	116
3.1 Nastavna tema: Ponavljanje	116
3.2 Nastavna tema: Grananje unutar programa	121
3.3 Nastavna tema: Tekstualne varijable	125
4. Gradivo četvrtog razreda	128
4.1 Nastavna tema: Ponavljanje	128
4.2 Nastavna tema: Grananje unatrag u programu ...	133

Predgovor tiskanom izdanju

Ova publikacija nastala je nakon izrade slične publikacije na Internetu. Dakle sadržaj koji je tiskan skoro je identičan sadržaju stranice

<http://www.magistra-nova.hr/pocetnica.html>

a do te stranice možete doći i ako u tražilicu upišete dvije riječi: **početnica** i **Mema**.

Za tiskano izdanje smo se odlučile nakon posjete nekoliko osnovnih škola u Primorsko-goranskoj županiji, gdje smo u razgovoru sa učiteljicama i učiteljima ustanovile da bi njima u radu koristilo tiskano izdanje iako postoji i elektroničko izdanje. Za tu ideju smo posebno zahvalne ravnateljici osnovne škole u Kraljevici, gospodinu Zrinku Tijanu.

Uvod

Osobno sam počela programirati nakon studija teoretske matematike na PMF-u kao programer pripravnik u Inu prije 45 godina. U ono vrijeme bila je sreća da sam i onda počela, jer se onda ljudi nisu time zamarali. Ipak, svjesna sam da je bolje početi ranije. Pitanje je koliko ranije? Odgovor je čim ranije, onda kada se i druge stvari uče, primjerice računati, čitati, raspoznavati boje, pisati, strani jezici i slično. Djeca koja su kao malena naučila strani jezik, igrati tenis ili skijati bolje to rade od svojih vršnjaka kad odrastu. S programiranjem je ista stvar, samo način na koji djeca uče mora biti prilagođen uzrastu djece. Zato za sasvim malu djecu ni jedan postojeći programski jezik nije dosta dobar. Za sasvim malu djecu oni su svi preteški. Djeca mogu učiti koristiti računalo, ali ne i programirati na nekom od postojećih jezika, čak ni ako se radi o Logu ili Basicu.

Pa ipak, učiti programirati može se učiti tako prirodno kako se pomoću slikovnica može učiti čitati, pomoću igre „Čovječe ne ljuti se“ brojati ili pomoću dječjih stihova Engleski. Autorica ovih redaka je prije četrdesetak godina razvila metodu pomoću koje su djeca mogla napisati program, provjeriti kako program radi, naučiti koliko je važan redoslijed naredbi, kako se ponašaju varijable u memoriji računala, a sve su to radila pomoću igre i igračke koju su sami izrađivali. Igračku smo nazvali „Mema“. Nastavu o toj tematici mogla im je držati njihova učiteljica koja o računalima nije morala znati ništa ili skoro ništa. Roditelji takva nastava nije koštala ni Kune, odnosno u ono vrijeme dinara. Djeca su naučila mnogo toga o *programiranju*. Tako su naučili da postoji *program*, da je on sastavljen od *naredbi*, da je *redoslijed* naredbi važan, da neke naredbe učitavaju brojeve u *memoriju*, da neke druge računaju sa *varijablama* ili brojevima (*konstantama*), da broj može biti tako velik, da ne stane u varijablu (overflow) da novi sadržaj varijable zamjenjuje stari...

Spontano su nadograđivali znanje kako se u neke varijable *kumuliraju* vrijednosti, da dva računala mogu biti *nekompatibilna*,

Ako sam vas makar malo zaintrigirala to je odlično. Možda ću obrisati prašinu sa starih papira i igračaka i možda ću za vas ponovo oživiti priču koja me je i prije toliko godina fascinirala. Ako nisam pokušati ću sa još nečim.

Metodu koju spominjem pokušala sam oživiti u jednoj riječkoj školi (Osnovna škola „Nikola Tesla“) prije više od trideset godina.

Učiteljica (Maja Mulac) je bila oduševljena, djeca također. Iskustva sam opisala u par stručnih i znanstvenih radova i osim toga nije se dogodilo skoro ništa. Ali ipak, nakon 6-7 godina nazvao me je nastavnik (Damir Čović) koji je u spomenutoj školi držao tehnički odgoj i u sklopu toga programiranje u Basicu za osme razrede. On mi je rekao da učenici iz jednog od četiri osma razreda puno lakše savladavaju Basic, nego ostala tri. U dogovoru s njim provela sam istraživanje u sva četiri razreda. Rezultat je bio fantastičan! Znanje programiranja u razredu koji je po prije spomenutoj metodi radio nekoliko sati u prvom razredu bilo je statistički značajno bolje od preostala tri razreda!. Što bi tek bilo da su nastavili sa radom kontinuirano!

Opet je taj rezultat bio objavljen u obliku znanstvenog rada i, pogađate, iza toga ništa. A mislim da bi iza toga moglo biti svašta i to puno toga lijepoga i dobrog. Možda sam radi toga pokušala još jednom, ovim stranicama. Na to me motivirala suradnja sa profesoricom Snježanom Babić, mojom bivšom studenticom i činjenica da se i na razini Europe javila inicijativa koja u mojem srcu nekad plamti, a nekad tinja već više od trideset godina.

Marina Čičin-Šain

Opis metode

„*Početnica programiranja*“ je metoda namijenjena učitelju ili učiteljici, koji su voljni učenike osnovne škole pripremiti za programiranje. Važno je istaknuti sljedeće: Učitelj ili učiteljica pri tom ne moraju znati mnogo o računalima, zapravo nije potrebno nikakvo predznanje.

Zamišljeno je da oni skupa sa svojim učenicima uče, korak po korak.

Stranice ovog priručnika trebale bi učiteljima i učiteljicama u tome pomoći. Tu će naći gradivo i upute, sat po sat kako mogu raditi sa djecom. Nove tehnologije omogućuju i da nam postavite pitanje kad nešto nije jasno ili da nas upozore ako smo u nečemu pogriješile.

Na poveznici: <http://www.magistra-nova.hr/pocetnica> može se pronaći e-mail adresa i poveznice na druge resurse naše zajednice.

Bit će nam drago da nas obavijestite kako napredujete u svom radu, da nam pošaljete koju fotografiju vaših učenika i učenica dok rade, koji crtež koji su izradili, neku njihovu primjedbu koju smatrate zanimljivom i sl. Već sama činjenica da razmišljate o tome da radite po ovoj metodi dovoljna je da nam se javite s eventualnim primjedbama.

Pri radu sa učiteljima ili učiteljicama znale smo naići na pitanje: “*A što ako mi učenik ili učenica postave pitanje na koje ne znam odgovoriti?*” Ne treba strahovati od takvih pitanja, ona su sasvim prirodna. Djeca puno puta postavljaju pitanja na koja odrasli nemaju odgovor. Po našem mišljenju takva pitanja i takvi učenici ili učenice su prava blagodat; zahvaljujući njima i sami ćemo uvijek biti prisiljeni učiti. Nemojte se pretvarati da znate, pohvalite pitanje i pokušajte naći odgovor na njega do slijedećeg sata. *Ako ne uspijete*, prosljedite nam pitanje, nastojati ćemo vam pomoći ponudivši odgovor. *Ako uspijete*, također nam pošaljite i pitanje i odgovor, na taj način također možete pomoći svojim kolegama i kolegicama ako jednog dana naiđu na sličan problem. Metoda u ovoj fazi ne zahtjeva nikakvu opremu. Koristit će se igračka „Mema”, koju će djeca sama na satu izraditi od stvari, kao što su: bojice, ljepilo, papir i slično.

Djeca neće biti dodatno opterećena, učiti će ono što ionako uče u prvom razredu: zbrajati, raspoznavati boje, crtati, razvijati motoriku. Pri tome će sasvim usput čuti za neke pojmove iz informatike ne na razini definicija, nego upravo čemu te stvari služe.

Ti pojmovi su: program, naredba, varijabla, redosljed naredbi, korisnik, ulaz, obrada, izlaz, podaci, konstante, kompatibilnost, kapacitet polja, adresa i još mnogi drugi.

Kako? To ćete doznati radom iz sata u sat.

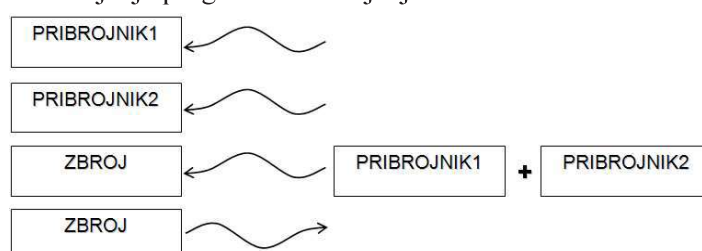
Kada se kasnije učenici susretnu s pravim programiranjem na računalu, njima će mnoge stvari biti poznate i lakše će usvajati novo gradivo. Neke stvari koje prvašići lako i spontano usvajaju pomoću “**Meme**” studentima na fakultetu predstavljaju problem, jer nisu razvili odgovarajući način razmišljanja.

Marina i Snježana

3. Gradivo trećeg razreda

3.1 Nastavna tema: Ponavljanje

Zadatak 1: Ponavljanje programa za zbrajanje.



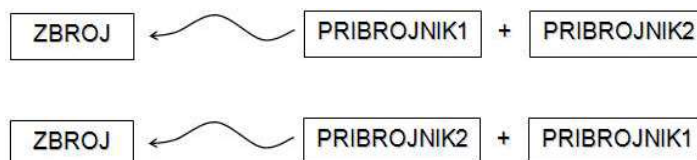
Izvrši program ako su zadani brojevi:

PRIBROJNIK1	25	37	44	69	0	28	
PRIBROJNIK2	3	6	11	25	17	0	
ZBROJ							

Napomena: U posljednjem stupcu potrebno je upisati proizvoljno odabrana dva broja.

Pitanje: Smiju li naredbe za ulaz u PRIBROJNIK1 i ulaz u PRIBROJNIK2 zamijeniti mjesta? (Odgovor: Mogu. Jer je svejedno da li smo najprije učitali PRIBROJNIK1 ili PRIBROJNIK2. Međutim, važno je da i PRIBROJNIK1 i PRIBROJNIK2 učitamo prije naredbe u kojoj se računa zbroj. Dakle, važno je da ulazne varijable učitamo prije naredbe za obradu.)

Pitanje: Smije li u naredbi u kojoj se računa ZBROJ umjesto naredbe:

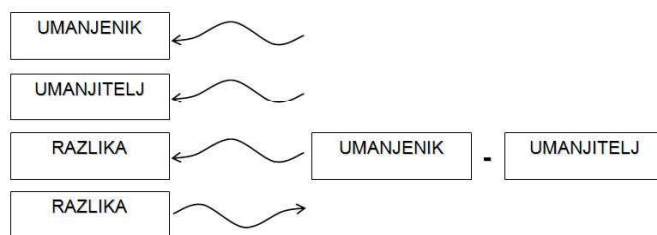


biti naredba:

(Odgovor: Da. Jer kod zbrajanja vrijedi da kod zbrajanja pribrojnici smiju zamijeniti mjesta.)

Pitanje: Smiju li naredbe za zbrajanje i ispis zbroja (3. i 4. naredba) zamijeniti mjesta? (Odgovor: Ne smiju. Jer je važno da najprije izračunamo ZBROJ, a zatim ga ispišemo.)

Zadatak 2: Ponavljanje programa za oduzimanje.



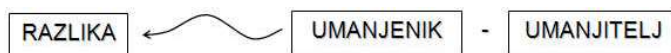
Izvrši program ako su zadani brojevi:

UMANJENIK	48	56	24	28	72		
UMANJITELJ	7	7	0	19	38		
RAZLIKA							

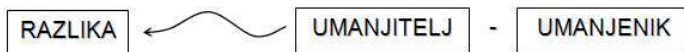
Napomena: U posljednjem stupcu potrebno je upisati proizvoljno odabrana dva broja.

Pitanje: Smiju li naredbe za ulaz u UMANJENIK i ulaz u UMANJITELJ zamijeniti mjesta? (Odgovor: Mogu. Jer je svejedno da li smo najprije učitali UMANJENIK ili UMANJITELJ. Važno je da ulazne varijable učitamo prije naredbe za obradu.)

Pitanje: Smije li u naredbi u kojoj se računa RAZLIKA umjesto naredbe:



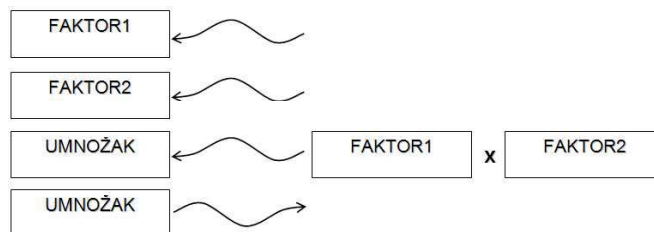
biti naredba:



(Odgovor: Ne. Jer kod oduzimanja uvijek umanjitelja oduzeti od umanjenika.)

Pitanje: Smiju li naredbe za računanje razlike i ispis razlike (3. i 4. naredba) zamijeniti mjesta? (Odgovor: Ne smiju. Jer je važno da najprije izračunamo RAZLIKU, a tek nakon toga je možemo ispisati.)

Zadatak 3: Ponavljanje programa za množenje.



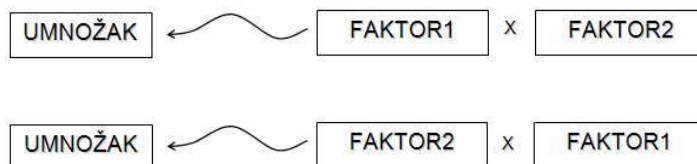
Izvrši program ako su zadani brojevi:

FAKTOR1	2	7	6	10	0	1		
FAKTOR2	3	4	10	4	8	4		
UMNOŽAK								

Napomena: U posljednjem stupcu potrebno je upisati proizvoljno odabrana dva broja.

Pitanje: Smiju li naredbe za ulaz u FAKTOR1 i ulaz u FAKTOR2 zamijeniti mjesta? (Odgovor: Mogu. Jer je svejedno da li smo najprije učitali FAKTOR1 ili FAKTOR2. Važno je da ulazne varijable učitamo prije naredbe za obradu.)

Pitanje: Smije li u naredbi u kojoj se računa UMNOŽAK umjesto naredbe:

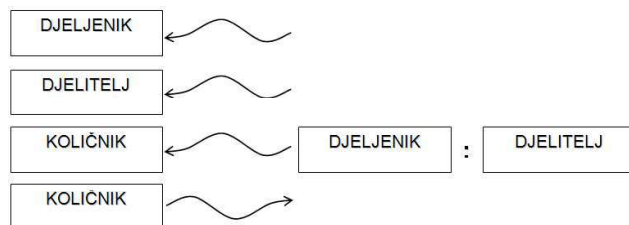


biti naredba:

(Odgovor: Da. Jer kod množenja vrijedi da faktori smiju zamijeniti mjesta.)

Pitanje: Smiju li naredbe za računanje umnoška i ispis umnoška (3. i 4. naredba) zamijeniti mjesta? (Odgovor: Ne smiju. Jer je važno da najprije izračunamo UMNOŽAK, a tek nakon toga ga možemo ispisati.)

Zadatak 4: Ponavljanje programa za dijeljenje:



Izvrši program ako su zadani brojevi:

DJELJENIK	12	15	72	42	63	0	5		
DJELITELJ	2	3	8	7	9	4	1		
KOLIČNIK									

Napomena: U posljednja dva stupca potrebno je upisati proizvoljno odabrana dva broja.

Pitanje: Smiju li naredbe za ulaz u DJELJENIK i ulaz u DJELITELJ zamijeniti mjesta? (Odgovor: Mogu. Jer je svejedno da li smo najprije učitali DJELJENIK ili DJELITELJ. Važno je da ulazne varijable učitamo prije naredbe za obradu.)

Pitanje: Smije li u naredbi u kojoj se računa RAZLIKA umjesto naredbe:



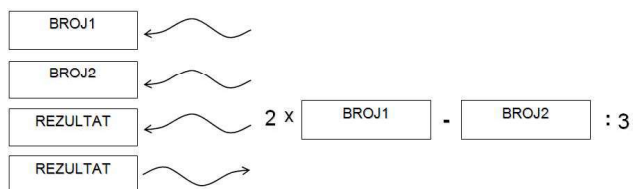
biti naredba:

(Odgovor: Ne. Jer kod oduzimanja uvijek umanjitelja oduzeti od umanjenika.)

Pitanje: Smiju li naredbe za računanje količnika i naredba za ispis količnika (3. i 4. naredba) zamijeniti mjesta? (Odgovor: Ne smiju. Jer je važno da najprije izračunamo KOLIČNIK, a tek nakon toga ga Važno je da ulazne varijable učitamo prije naredbe za obradu možemo ispisati.)

Zadatak 5: Ponavljanje složenijih zadataka:

Pitanje: Što će se ispisati iz polja (pretinca) REZULTAT nakon izvršenog programa?

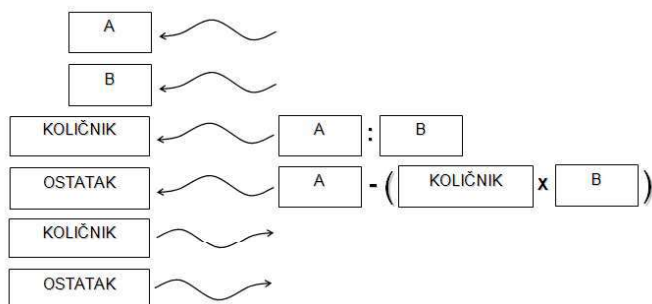


Nadopuni tablicu:

BROJ1	2	8	10	10	5		
BROJ2	3	6	30	15	30		
REZULTAT							

Napomena: U posljednja dva stupca potrebno je upisati proizvoljno odabrana dva para brojeva.

Zadatak 6: Dijeljenje (s ostatkom):



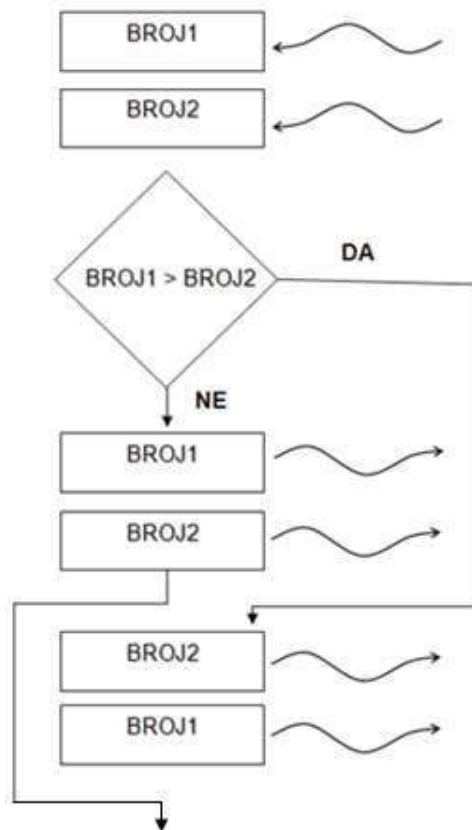
Provjeri rad programa s ulazima:

A	7	12	29	37	24	0	93	37	
B	2	5	10	7	3	5	3	1	
KOLIČNIK	3								
OSTATAK	1								

Napomena: U posljednjem stupcu potrebno je upisati proizvoljno odabrana dva broja.

3.2 Nastavna tema: Grananje unutar programa

Zadatak: Što radi ovaj program?



Odgovor: Program učitava dva broja. Ako su brojevi različiti ispisuje manjeg pa većeg. Ako su brojevi jednaki ispisat će dva puta isti broj. To se može ustanoviti tako da najprije zadate učenicima da provjere rad programa sa parovima brojeva kao u tablici koja slijedi. Za prvi par provjeru radite svi zajedno.

Unijeli smo brojeve: 3 i 2; 14 i 21; 27 i 35; 77 i 59. Provjeri rad programa.

Pitanje: Što se ispisalo?

UPUTA ZA NASTAVNIKA:

U ovom programu po prvi put se naredbe ne odvijaju redom nego imamo i preskok.

Preskok se događa nakon ispitivanja i ovisno o odgovoru na postavljeno pitanje nastavljamo ili sa sljedećom naredbom ili s naredbom koja se nalazi negdje kasnije

u programu. Slijed naredbi dan je ravnim **strelicama**, a postupak se zove **grananje**.

Zadatak: Provjeri što se događa korak po korak u programu.

Npr. ako smo unijeli 3 i 2 znači da je 3 ušao u varijablu BROJ1, a 2 u BROJ2. Rezultat usporedbe $BROJ1 > BROJ2$ je „DA“, jer 3 je veći od 2.

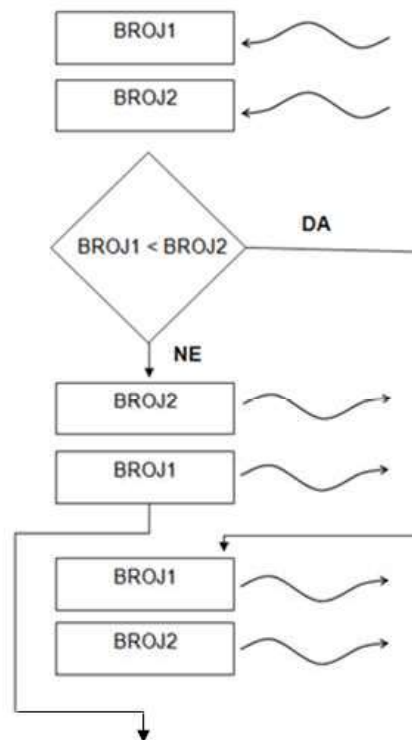
Zato imamo grananje, pa se ispisuje najprije sadržaj BROJ2, što je 2, a iza toga BROJ1, dakle 3. I program završava s radom.

Ako smo unijeli 14 i 21 znači da je 14 ušlo u BROJ1, a 21 u BROJ2. Rezultat usporedbe dva broja ($BROJ1 > BROJ2$) u ovom slučaju je „NE“ pa se nastavlja s naredbom u slijedu, dakle, ispisuje se sadržaj iz pretinaca BROJ1 pa iz pretinca BROJ2. Nakon toga nužno je da se preskoči ispis za uvjet koji nije ispunjen i program završava s radom.

Zadatak: Ponovi provjeru za ostale parova brojeva.

Pitanje: Što će se dogoditi ako unesemo par istih brojeva? Da li se taj isti program mogao napisati na drugi način?

Rješenje:



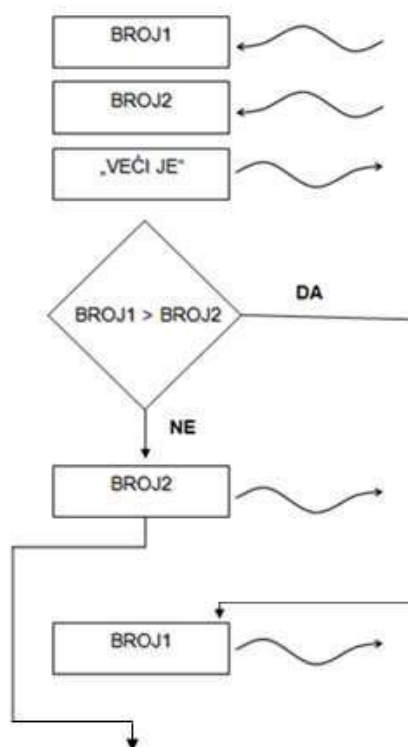
Zadatak: Provjeri rad programa. Ako smo unijeli parove: 14 i 22; 27 i 16; 34 i 58; 43 i 11; 35 i 35. Što se ispisalo?

UPUTA ZA NASTAVNIKA:

Provjeri što se događa korak po korak. Učenici trebaju izgovoriti sve što se događa, a mogu napisati samo ono što se ispisuje.

Primjer: USPOREDBA DVA BROJA

Zadatak: Što radi ovaj program?



Odgovor: U ovom programu imamo novost da u trećoj naredbi ispisujemo konstantnu vrijednost odnosno tekst “VEĆI JE”.

Zadatak: Provjeri što radi ovaj program ako su u pretince ušli parovi brojeva: 12 i 3; 225 i 36; 77 i 181; 116 i 611.

Primjer: Za prvi par BROJ1 je 12, a BROJ2 je 3 ispisat će se najprije tekst „VEĆI JE“, a zatim će se usporediti da li je $12 > 3$. Zato jer je odgovor „DA“, ispisat će se BROJ1, odnosno 12.

Dakle, rezultat programa je ispis:

Veći je 12

Na isti način provjeri i za ostale parove, tako da se izgovori što sve program radi, a napiše što program ispisuje.

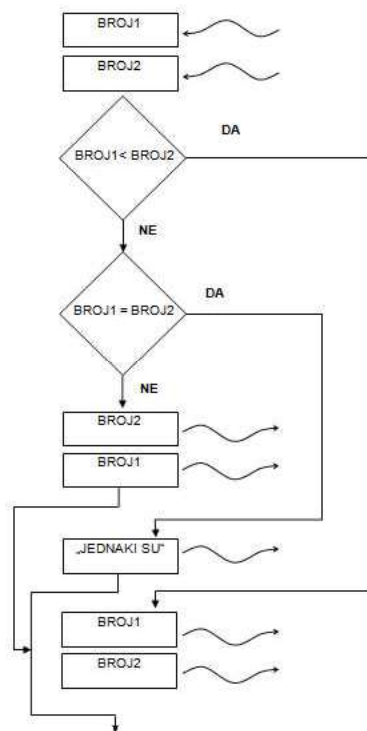
Pitanje: Što će se dogoditi ako se unese par istih brojeva? **Odgovor:** Ispisat će se jedan od tih istih brojeva bez ikakve napomene.

Npr. za par 4 i 4 ispisat će se:

Veći je 4

Ako želimo da se u tom slučaju ispiše npr. komentar „BROJEVI SU JEDNAKI” trebamo napisati novi, složeniji program koji to rješava.

Zadatak: Unesi dva broja. Programi provjeravaju koji je broj manji, a koji veći i ispisuje manjeg pa većeg osim ako su jednaki. Ako su jednaki ispisuje poruku: „BROJEVI SU JEDNAKI”.

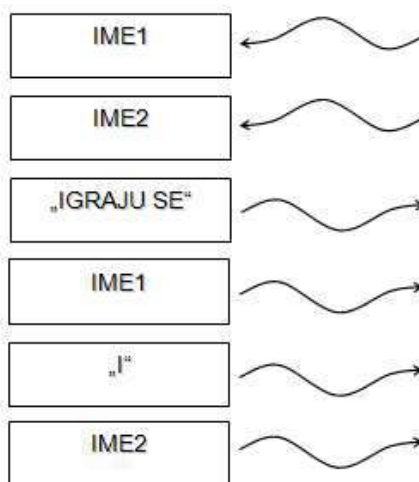


3.3 Nastavna tema: Tekstualne varijable

Do sada smo koristili tekstualne konstante koje su se nalazile u programu. Tekst može biti i sadržaj pretinaca.

Varijable (pretinci) se mogu puniti i tekстом odnosno znakovima, ali se onda s njima ne može računati. Te varijable mogu se ispisivati i međusobno uspoređivati.

Zadatak: Što radi ovaj program?



Uputa: Ako u IME1 unesemo IVO, a u IME2 ANTE onda će se ispisati:

```
IGRAJU SE
IVO
I
ANTE
```

Pitanja:

1. Što će se ispisati ako unesemo par imena ANA, IVO?
2. Što će se ispisati ako unesemo DRVO, ČAŠA?
3. Što će se ispisati ako unesemo 2, 3?
4. Što će se ispisati ako unesemo +, - ?
5. Što zaključujete?

Primjer: Što radi ovaj program?



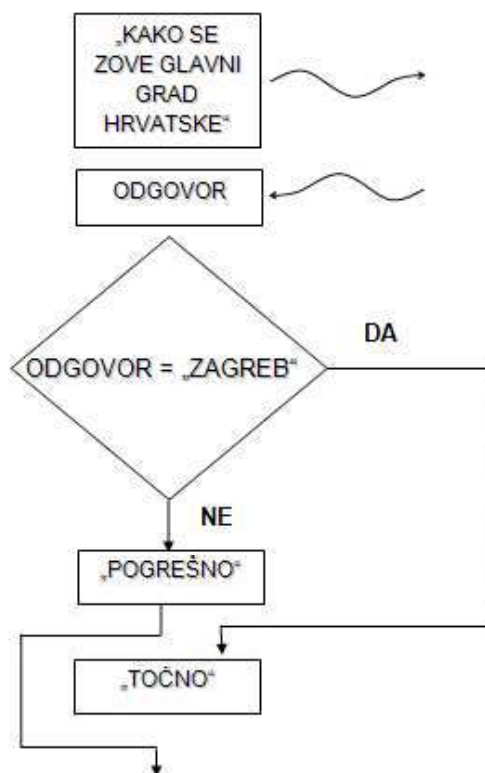
Odgovor: Program najprije korisniku postavi pitanje “KAKO SE ZOVEŠ”, zatim od korisnika očekuje odgovor koji smješta u varijablu IME. Ako korisnik npr odgovori: ANA to će postati sadržaj varijable IME. Nakon toga računalo ispisuje:

```
DOBAR DAN
ANA
DRAGO MI JE DA SMO SE UPOZNALI
```

Tako se može isprogramirati razgovor s računalom.

Ranije smo rekli da se tekst može provjeriti ako je jednak. To ćemo koristiti u sljedećem zadatku.

Zadatak: Što radi sljedeći program?



Pitanje:

1. Ako u ODGOVOR unesemo RIJEKA što će se ispisati? („POGREŠNO“)
2. Što treba unijeti u ODGOVOR da bi se ispisao tekst „TOČNO“? (ZAGREB)
3. Možemo li u ODGOVOR upisati IVO? (Možemo, ali će ispis biti „POGREŠNO“)
4. Možemo li ODGOVOR upisati 48? (Možemo, ali će ispis biti „POGREŠNO“)