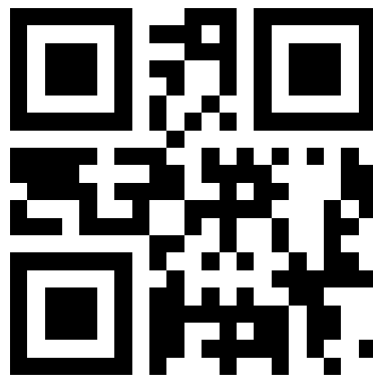




RAČUNARSTVO I INFORMATIKA – ISM

ČAS 1



Literatura:

- Milan Čabarkapa, Računarstvo i informatika, Beograd 2004
- Milan Čabarkapa, Nevenka Spalević, Metodička zbirka zadataka iz programiranja

Materijali sa časova će biti postavljeni na:

<https://imi.pmf.kg.ac.rs/moodle/course/view.php?id=296>

Kontakt: marina.svicevic@gmail.com



Rešavanje problema korišćenjem kompijuterskog programa može se podeliti na više segmenata:

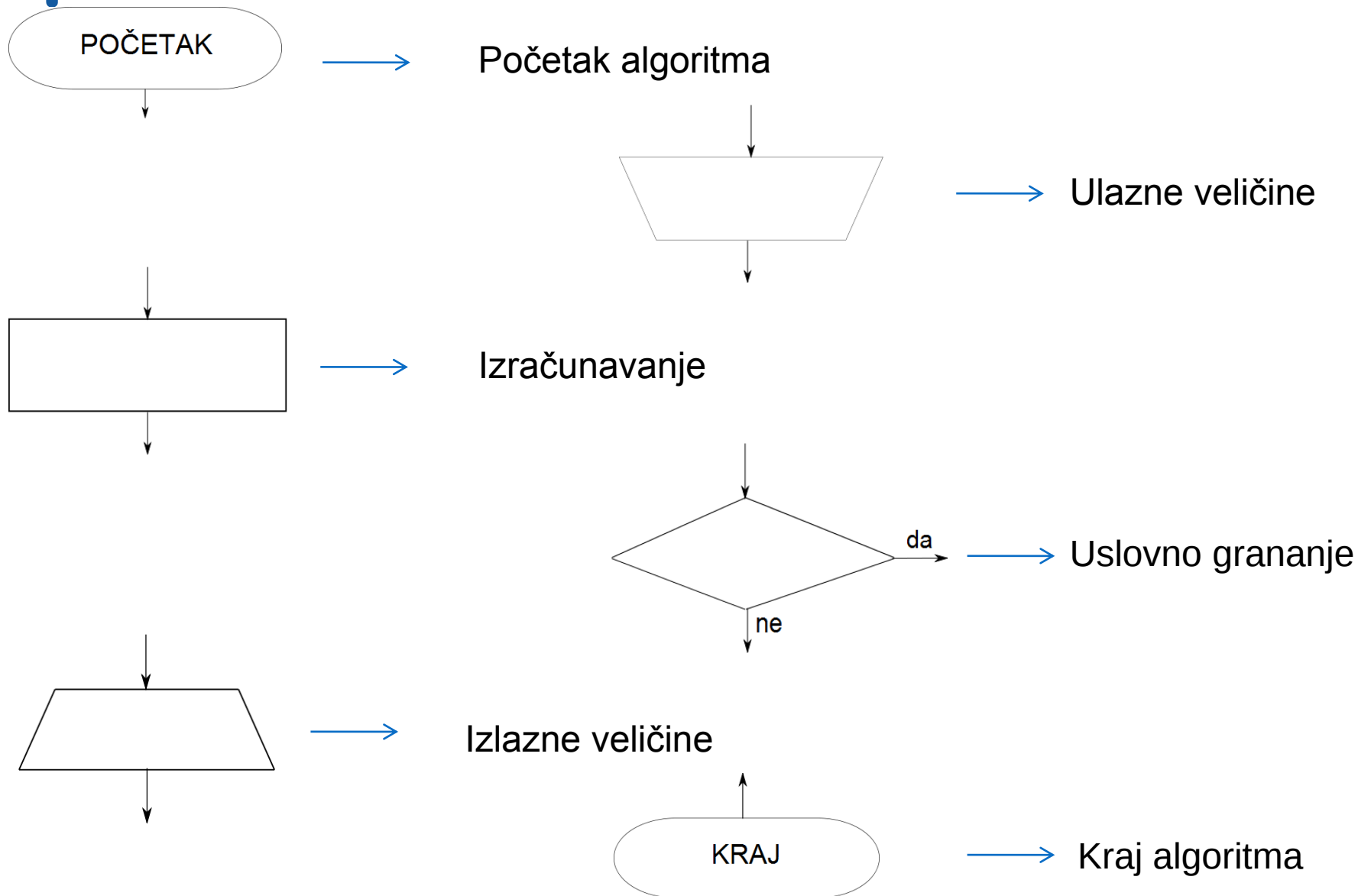
- DEFINISANJE PROBLEMA
- ANALIZA PROBLEMA
- CRTANJE ALGORITMA
- PROJEKTOVANJE PROGRAM
- KODIRANJE
- TESTIRANJE
- ANALIZA REZULTATA
- ISPORUKA PROGRAMA
- ODRŽAVANJE



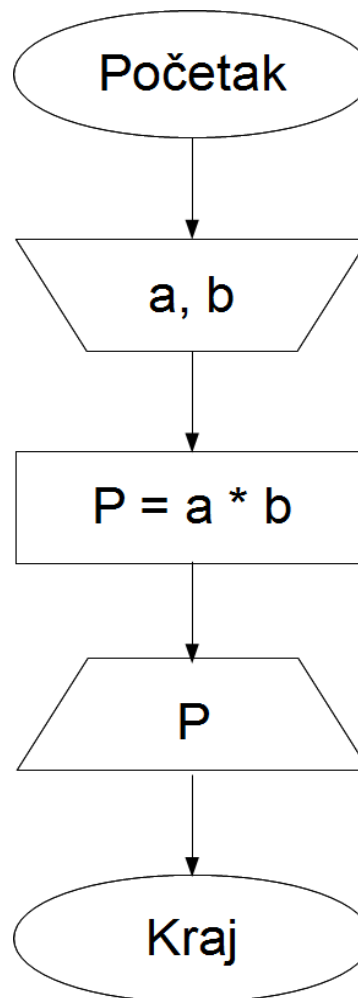
- **ALGORITAM** je konačan je skup nekih pravila ili koraka po kojima se rešava određeni zadatak.
- Svaki algoritam se može opisati na prirodnom jeziku (srpski, engleski...)
- Zadatak: Kuvanje kafe
Rešenje (algoritam):
 1. Sipati vodu u lonče za kafu.
 2. Staviti lonče na ringlu.
 3. Uključiti ringlu.
 4. Sačekati nekoliko minuta dok voda ne proključa.
 5. Sipati kašičicu kafe u vruću vodu i promešati.
 6. Sačekati nekoliko sekundi dok nivo vode ne počne da raste.
 7. Skloniti lonče sa ringle.
 8. Isključiti ringlu.
 9. Sipati kafu iz lončeta u šoljicu.



- Ovakvo pisanje algoritma je dosta tesko i nečitko
- Češće se koristi grafičko predstavljanje pomoću *algoritamske šeme*.
- Algoritamska šema se može sastojati od sledećih grafičkih simbola

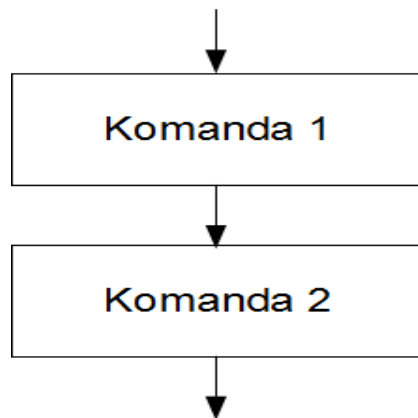


Napisati algoritam koji za date dimenzije pravougaonika računa njegovu površinu.

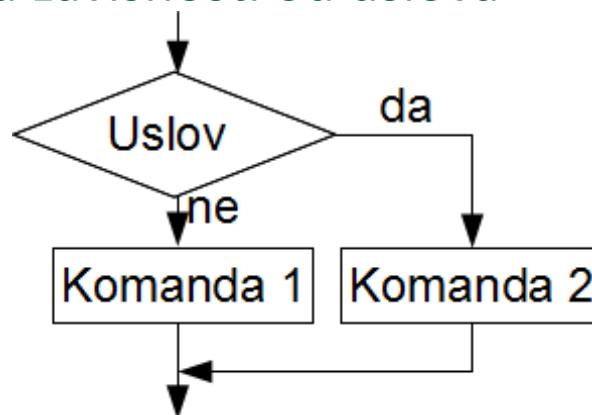




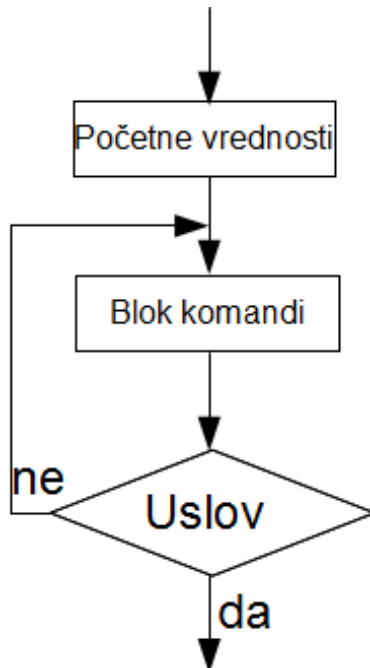
- Svaki algoritam se može predstaviti korišćenjem tri osnovne strukture
 - *Linijaska struktura** – komande se izvršavaju jedna za drugom



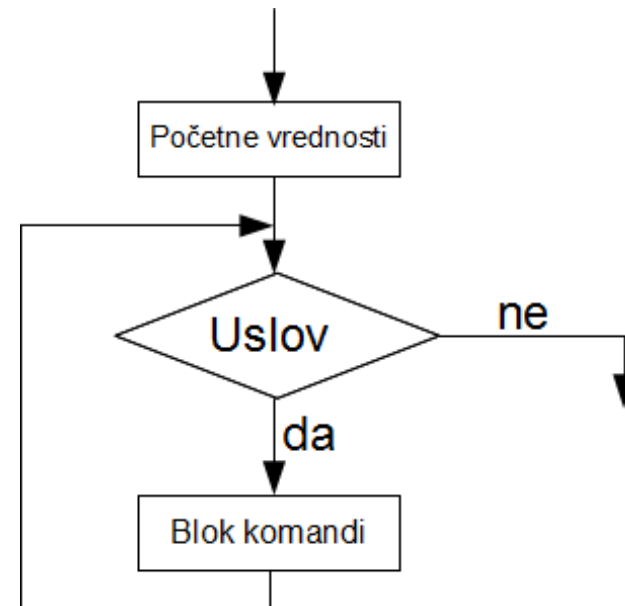
- *Razgranata struktura** - omogućava granjanje izvršenja programa u zavisnosti od uslova



***Ciklična struktura** - niz komandi koje se mogu izvršavati uzastopno više puta u zavisnosti od ispunjenosti uslova



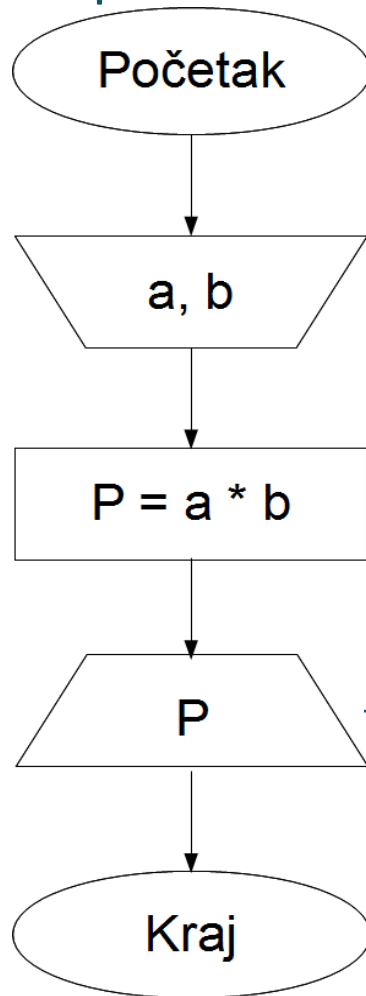
Kada se uslov nalazi na kraju ciklusa, blok naredbi se izvršava bar jednom



Kada se uslov nalazi na početku ciklusa, blok naredbi ne mora da se izvrši ni jednom



Napisati algoritam koji za date dimenzije pravougaonika računa njegovu površinu.



```
Program prvi;  
var a,b,P: integer ;  
begin  
  read(a,b);  
  P:= a*b;  
  write(P);  
end.
```



Elementi programskog jezika pascal

- Sve konstrukcije u programskom jeziku Pascal se grade od skupa osnovnih simbola jezika koji čine slova, cifre i specijalni znaci.

Slova

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z

Cifre

0	1	2	3	4	5	6	7	8	9
---	---	---	---	---	---	---	---	---	---

Specijalni znaci

+	-	*	/	=	<	>	[	]	.	,	:	;	^	(	)	'	{	}
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---



- Osnovni elementi jezika su i **rezervisane reči** koje imaju unapred definisano značenje i ne mogu se koristiti u druge svrhe.

and	array	begin	case	const	div	do
downto	else	end	file	for	function	goto
if	in	label	mod	nil	not	of
or	packet	procedure	program	record	repeat	set
then	to	type	until	var	while	with

- Većina PASCAL kompajlera ne pravi razliku između malih i velikih slova, sam programer bira kojim stilom će pisati program. Najvažnije je da program bude čitak i uredan.



- Podacima se naziva sve ono što može da bude predmet obrade računarom ili se može dobiti kao rezultat obrade.
- Svi tipovi podataka predstavljeni su u binarnom obliku tj. kao niska nula i jedinica.
- Kompajleri prepoznaju određene tipove podataka i prevode ih u binarni oblik po određenim pravilima, a prilikom čitanja iz memorije kompajler prevodi podatke iz binarnog oblika u odgovarajući tip višeg programskog jezika.



- Tip podatka određuje skup vrednosti koje podatak može da ima.
- U PASCAL-u tipovi podataka se mogu podeliti u nekoliko grupa:
 - Prosti tipovi
 - Složeni tipovi
 - Pokazivači

Osnovni tipovi podataka

- ⦿ integer 9, 76, 15
- ⦿ real 3.75 , 6.78
- ⦿ char 'a' , 'y'
- ⦿ string 'summer'
- ⦿ boolean TRUE, FALSE



Celobrojni tip - integer

- Za rad sa celim brojevima se koristi **celobrojni tip podataka**. U PASCAL-u se koristi rezervisana reč **integer**.
- Znak + ispred broja se podrazumeva, tako da u opstem slučaju ne mora da se pise.

Realni tip - real

- U programskom jeziku PASCAL-u realni brojevi se definišu rezervisanom rečju **real**.
- Prilikom pisanja decimalnog broja celobrojni i razlomljeni deo se odvajaju **decimalnom tačkom**.

Logički tip - boolean

- Logički tip se koristi za ispitivanje istinitosti nekog logičkog iskaza. Može imati dve vrednosti **TRUE** i **FALSE** tj. **tačno** i **netačno**.

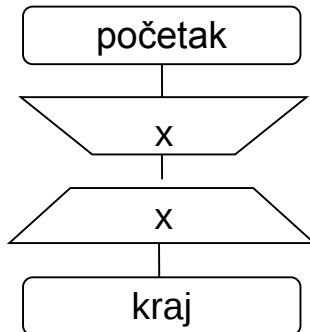


Znakovni tip - char

- Znakovni tip podataka je uređeni skup znakova koji mogu biti:
 1. Slova abecede
 2. Numerički znakovi (od 0 do 9)
 3. Znakovi interpunkcije
 4. Specijalni znakovi
- Označava se rezervisanom rečju **char**.
- Svakom znaku pridružen je po jedan ceo broj – **kod**.
- Najčešće se koristi ASCII tabela
- Znak:=‘b’ – apostrofi su obavezni

Nizovi znakova (reči i rečenice) nazivaju se stringovima i označavaju se rezervisanom rečju **string**.

Napisati algoritam i program koji učitava **celobrojnu** promenljivu x i na standardnom *Output*-u ispisuje istu!



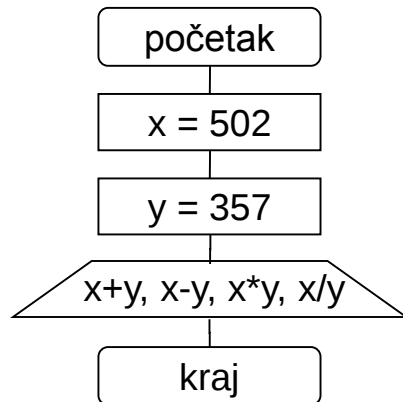
```

program zad_1;
var x:integer;
begin
  readln(x);
  writeln(x);
end.
  
```

```

program zad_1;
var x:integer;
begin
  write('Unesi promenljivu x:');
  readln(x);
  writeln('x = ',x);
end.
  
```

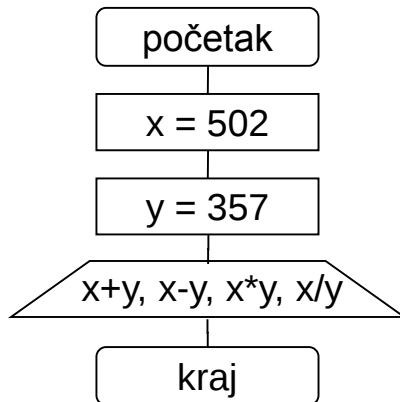
Za dva pozitivna cela broja ispisati njihov zbir, razliku, proizvod i količnik.



```
program zad_2;  
var x,y:integer;  
begin  
    x:= 502;  
    y:= 357;  
    write(x+y, x-y, x*y, x/y)  
end.
```

Za dva pozitivna cela broja ispisati njihov zbir, razliku, proizvod i količnik.

– formatiranje ispisa



```
program zad_3;
var x,y:integer;
begin
    x:= 502;
    y:= 357;
    writeln(x+y:10, x-y:10, x*y:10, x/y:10:2)
end.
```



Za dva pozitivna cela broja ispisati njihov zbir, razliku, proizvod i količnik.

– deklarisanje promenljive za rezultat – **integer** tip

```
program zad_4;  
var x,y,r1,r2,r3,r4:integer;  
begin  
  x:= 502;  
  y:= 357;  
  r1:= x + y;  
  r2:= x - y;  
  r3:= x * y;  
  r4:= x / y;  
  writeln(r1:10, r2:10, r3:10, r4:10)  
end.
```



Za dva pozitivna cela broja ispisati njihov zbir, razliku, proizvod i količnik.

– deklarisanje promenljive za rezultat – **real** i **integer**
tip

```
program zad_5;  
var x,y,r1,r2,r3:integer;  
    r4:real;  
begin  
    x:= 502;  
    y:= 357;  
    r1:= x + y;  
    r2:= x - y;  
    r3:= x * y;  
    r4:= x / y;  
    writeln(r1:10, r2:10, r3:10, r4:10:2)  
end.
```



Za dva pozitivna cela broja ispisati njihov zbir, razliku, proizvod i količnik.

- deklarisanje promenljive za rezultat – real, integer i longint tip

```
program zad_6;
var x,y,r1,r2:integer;
    r3:longint;
    r4:real;
begin
  x:= 502;
  y:= 357;
  r1:= x + y;
  r2:= x - y;
  r3:= x * y;
  r4:= x / y;
  writeln(r1:10, r2:10, r3:10, r4:10:2);
  writeln(maxint);
  writeln(maxLongint);
end.
```