

Osnove pseudo jezika

Pri opisu programskih struktura bit će korištene osnove pseudo jezika koji se rabi u *Ispitnom katalogu za državnu maturu iz Informatike*



Varijable

- To je **mjesto u memoriji** rezervirano **za pohranu podatka**.
- Varijablu jednoznačno određuje **ime** (identifikator).
- Ime varijable se u pseudo jeziku može **zadati proizvoljno**, npr. :
 - x, a, TX, B_Pod, AmPe

2

Vrijednost varijable

- Varijabli se vrijednost pridružuje s pomoću **operatora pridruživanja**. Za ovu inačicu pseudo jezika to je **:=**.
Npr. izraz **x:=3** se može čitati "varijabli x se pridružuje broj 3".
- **Svaka naredba** ove inačice pseudo jezika završava znakom točka-zarez (;).

3

Operator pridruživanja

- Kolika će biti vrijednost varijable **X** na kraju ovog odsječka?
X := 5;
X := 3;
X = 5;

Rezultat: **X** ima vrijednost **3**

4

operatori

- To su **simboli** koji predstavljaju određene **funkcije**.
- Može ih se svrstati u skupine, npr.:
 - **aritmetički** operatori,
 - **logički** operatori,
 - operatori uspoređivanja (**relacijski** operatori).

5

Aritmetički operatori

funkcija	operator
• Zbrajanje	+
• Oduzimanje	-
• Množenje	*
• Dijeljenje	/
• Cjelobrojno dijeljenje	DIV
• Cjelobrojni ostatak dijeljenja	MOD

6

Operatori dijeljenja

$x := 5;$ $y := 2;$	$a := x / y;$
	$b := x \text{ DIV } y;$
	$c := x \text{ MOD } y;$

a je 2,5	b je 2	c je 1
----------	--------	--------

7

Logički operatori

- **Logički podaci** su podaci koji mogu poprimiti samo jednu od dvije vrijednosti, npr.: **1/0**.
- Za rad s logičkim podacima, postoje logičke funkcije.

funkcija	operator
• Logički I	I
• Logički ILI	ILI
• Logički NE	NE

8

Logičke funkcije

A	NE A
1	0
0	1

A	B	A I B
0	0	0
0	1	0
1	0	0
1	1	1

A	B	A ILI B
0	0	0
0	1	1
1	0	1
1	1	1

9

Logički operatori

$a := 0;$ $b := 1;$	$e := a \text{ I } b;$
	$f := a \text{ ILI } b;$
	$g := \text{NE } a;$

e je 0	f je 1	g je 1
--------	--------	--------

10

Operatori uspoređivanja

funkcija	operator
manje	<
manje ili jednako	<=
veće	>
veće ili jednako	>=
jednako	=
različito	<>

11

Operatori uspoređivanja

- Rezultat usporedbe je 1 ako je izraz zapisan operatorima usporedbe istinit, a ako nije, rezultat će biti 0.

$a := (5 < 13);$
$b := (8 <= 8);$
$c := (5 = 8);$
$d := (5 <> 2 + 3);$

a je 1
b je 1
c je 0
d je 0

12

Redoslijed izvršavanja operatora

()
NE
* / DIV MOD I
+ - Ili
> < <= >= <> =

13

Redoslijed izvršavanja operatora

x := 22 DIV 5 * 11 MOD 3;

- Operatori su ravnopravni, izraz se izvršava s lijeva u desno:

22 DIV 5 = 4

4 * 11 = 44

44 MOD 3 = 2 (ostatak dijeljenja 44/3)

x je 2

14

Redoslijed izvršavanja operatora

x := (22 DIV 5) * (11 MOD 3);

- Zagrade poništavaju prioritete operatora :

22 DIV 5 = 4

11 MOD 3 = 2

4 * 2 = 8

x je 8

15

Redoslijed izvršavanja operatora

x := 3+4*6/3-(55 MOD 6);

55 MOD 6 = 1

4 * 6 = 24

24 / 3 = 8

3 + 8 - 1 = 10

x je 10

16

Programska struktura

- Opisuje **način** i **redoslijed izvršavanja** pojedinih **radnji** koje dovode do konačnog rješenja zadatka.
- Razlikuju se osnovne programske strukture:
 - **pravocrtna** programska struktura (slijed, niz),
 - struktura **grananja**,
 - struktura **petlje**.

17

Naredbe pravocrtne strukture

opis	naredbe pseudo jezika
unos	<u>ulaz</u>
ispis	<u>izlaz</u>

- Iza naredbe, unutar para okruglih zagrada **navode se varijable** u koje će se pohraniti uneseni podaci ili čiji će se sadržaj ispisati, npr.:

ulaz(a,b);

izlaz(d);

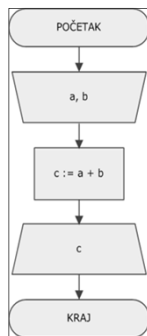
18

Pravocrtna programska struktura

- Korisnik unosi dva broja, brojevi se zbrajaju, a dobiveni se zbroj ispisuje.

```
ulaz (a, b);
c := a+b;
izlaz (c);
```

- Program se odvija uvijek istim slijedom naredbi bez obzira na unesene podatke.



19

Funkcije

- To su izdvojeni nizovi naredbi koji čine logičke cjeline a obavljaju točno utvrđene zadatke.

opis	naredbe pseudo jezika
Apsolutna vrijednost realnoga broja.	abs(x)
Drugi korijen realnoga broja.	sqrt(x)
Zaokruživanje realnoga broja na najbliži cijeli broj.	round(x)
Zaokruživanje na cijeli broj uz odbacivanje decimalnih znamenaka.	trunc(x)
Kvadrat realnog broja.	sqr(x)

20

Funkcije - primjer

- Potrebno je unijeti realni broj, a potom:

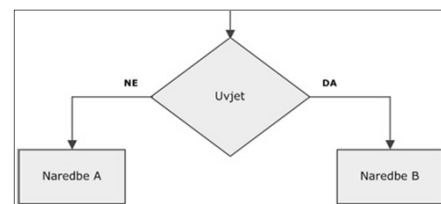
- Izračunati drugi korijen broja.
- Zaokružiti uneseni broj na najbliži cijeli broj.
- Izračunati kvadrat broja.
- Ispisati dobivene vrijednosti.

```
ulaz (x);
a := Sqrt(x);
b := Round(x);
c := Sqr(x);
izlaz (a, b, c);
```

21

Programska struktura grananja

- Grananje je programska struktura koja omogućuje različit tijek programa, ovisno o rezultatu postavljenog uvjeta.



22

Naredbe grananja

- Dijelovi programa koji se uvjetno izvode, grupiraju se u skupine ili blokove naredbi.

opis	naredbe pseudo jezika
grananje	ako je uvjet onda naredba1 inače naredba2;
blok naredbi	{ }

23

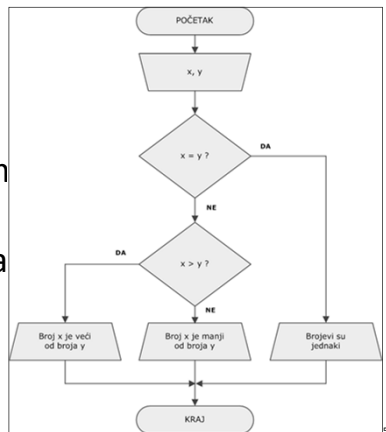
Programska struktura grananja

- Korisnik unosi dva broja, a zatim se određuje i ispisuje veći broj.

```
ulaz (x, y);
ako je x = y onda
    izlaz („Brojevi su jednaki”)
inače
{
    ako je x > y onda
        izlaz (x)
    inače
        izlaz (y);
}
```

24

Dijagram tijeka grananja



Programska struktura petlje

- Često je u programu potrebno neku radnju ponoviti više puta (iterirati). Takva se programska struktura naziva petlja.
- Ponavljati se može:
 - unaprijed zadani broj puta,
 - sve dok je ispunjen zadani uvjet.

26

Naredbe petlje

opis	naredbe pseudo jezika
Petlja s unaprijed <i>poznatim brojem ponavljanja</i>	<u>za</u> b := p <u>do</u> k <u>činiti</u> naredba;
Petlja kod koje <i>nije unaprijed poznat broj ponavljanja</i> , a uvjet se provjerava <i>na početku</i> petlje	<u>dok je</u> uvjet <u>činiti</u> naredba;
Petlja kod koje <i>nije unaprijed poznat broj ponavljanja</i> , a uvjet se provjerava <i>na kraju</i> petlje	<u>ponavljati</u> naredba; <u>dok je</u> uvjet;

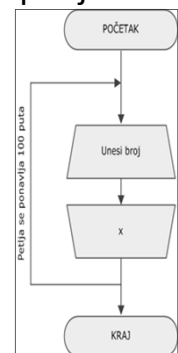
27

Programska struktura petlje

- Korisnik treba unijeti 100 brojeva, a prije svakog unosa broja ispisuje se tekst "Unesi broj".

```

za b := 1 do 100 činiti
{
  izlaz („Unesi broj");
  ulaz (x);
}
  
```



28