

SVEUČILIŠTE U ZAGREBU
FAKULTET PROMETNIH ZNANOSTI

RAČUNALSTVO

Prof. dr. sc. Hrvoje Gold

2009/2010.

RAČUNALSTVO

08. OSNOVE PROGRAMIRANJA

2

Uvod

- Program je postupak rješavanja zadatka prilagođen izvođenju na računalu
- Osnove programiranja izložene su pomoću programskog jezika Pascal

3

Elementi programa ...

- Ulazna operacija - unos vrijednosti s ulazne jedinice (tipkovnica, disk, ulazna vrata) u program
- Podaci u programu - konstante, varijable i strukture čuvaju vrijednosti:
 - brojeve (cijeli, decimalni),
 - tekstove (znakove, karaktere i nizove)
 - adrese (varijabli ili struktura)
 na određenom mjestu u memoriji (memorijskoj lokaciji)

4

Elementi programa ...

- Operacije u programu - rukovanje podacima pomoću naredbi, instrukcija:
 - pridruživanje vrijednosti,
 - slaganje vrijednosti (zbrajanje, dijeljenje, i sl.) i
 - uspoređivanje vrijednosti (jednako, različito i sl.)
- Izlazna operacija - iznos vrijednosti iz programa: ispis na izlaznu jedinicu – zaslon, tiskalo (riječi i slike), memorijsku jedinicu (disketa ili disk) ili izlazna vrata

5

Organizacija instrukcija

- Uvjetno izvođenje - instrukcija ili skup instrukcija izvodi se samo kada je zadovoljen određeni uvjet ili niz uvjeta
- Petlja - izvođenje instrukcije ili skupa instrukcija ponavlja se više puta
- Potprogram, procedura, funkcija - instrukcije se razlučuju u skupove kojima se pridružuje naziv, a izvode se pozivom odgovarajućeg skupa

6

Logička organizacija memorije

- Memoriju tvori niz adresibilnih memorijskih lokacija određene duljine
- Adresa memorije odgovara nazivu podatka (varijable, strukture)
- U memorijske lokacije pohranjuju se podaci: naredbe programa, vrijednosti, adrese drugih lokacija
- Duljina memorijske lokacije izražena je osnovnom informacijskom jedinicom (byte)

7

Tipovi podataka – cijeli broj

- Pohrana cjelobrojnih vrijednosti (integer):
 - u području od najmanje negativne vrijednosti do najveće pozitivne vrijednosti
 - područje vrijednosti ovisi o upotrijebljenom jezičnom prevoditelju

8

Pohrana cijelih brojeva

TIP	PODRUČJE VRIJEDNOSTI	DULJINA (byte)
byte (bajt)	0..255	1
shortint (kratki c.b.)	-128..127	1
integer (c.b.)	-32768..32767	2
word (riječ)	0..65535	2
longint (dugi c.b.)	-2147483648..2147483647	4

9

Tipovi podataka – realni broj

- Pohrana realnih vrijednosti (real)
 - u formatu pokretnog zareza sastavljenom od mantise (vrijednosti decimalnog broja) i eksponenta koji određuje mjesto zareza
 - Broj znamenaka mantise (značajne znamenke) i područje vrijednosti eksponenta ovise o korištenom jezičnom prevoditelju

10

Pohrana realnih brojeva

TIP	PODRUČJE VRIJEDNOSTI	ZNAČAJNE ZNAMENKE	DULJINA (byte)
real (decimalni broj)	$10^{-38}..10^{38}$	11	6
single (jednostruki)	$10^{-38}..10^{38}$	7	4
double (dvostruki)	$10^{-308}..10^{308}$	15	8

11

Tipovi podataka – znak

- Pohrana znakova (character)
 - pohrana slova abecede (a,b,..., A, b, ...)
 - znamenki (od 0 do 9)
 - simbola (? , ! . :)
- Pohrana jednog slova u jednom bajtu
- Znakovne konstante se prikazuju unutar navodnika ('a', 'E', '?', '2')
- Više znakova čini znakovni niz – string ('Osnove programiranja')

12

Tipovi podataka – Boole-ove konstante i kazala

- Booleove konstante: vrijednost istinitosti - istina ili laž
- Kazalo: pohrana adrese memorijske lokacije, a ne njezinog sadržaja

13

Identifikatori ...

- Nazivi konstanti, tipova podataka, varijabli, funkcija:
 - tipovi podataka određeni programskim jezikom - *integer, real*
 - naziv glavne funkcije programa i nazive varijabli zadaje programmer - *Telefon; impulsi, cijena*
 - procedure određene programskim jezikom - *readln, writeln*

14

Identifikatori

- Prvi znak identifikatora mora biti slovo
- Dopusćeni broj znakova identifikatora određuje jezični prevoditelj
- Nazivi tipova podataka, naredbi i procedura su rezervirane, ključne riječi programskog jezika

15

Operatori

- Rukovanje vrijednostima unesenima u program, odnosno pohranjenima u varijablama
- Služe za tvorbu izraza
- Razvoj izraza izvodi se s lijeva na desno
- Binarni (djeluju na dva operanda), unarni (djeluju na jednog operanda)
- Pridruživanja, usmjereni na rukovanje s bitovima, relacijski, logički, adresni, niza

16

Operator pridruživanja

- Rezultat razvoja izraza s desne strane operatora pridruživanja pridružuje se varijabli s lijeve strane operatora pridruživanja

17

Aritmetički operatori

ZBRAJANJE	+	$7 + 2 = 9$
ODUZIMANJE	-	$7 - 2 = 5$
MNOŽENJE	*	$7 * 2 = 14$
DIJELJENJE	/	$7 / 2 = 3.5$
CJELOBROJNO DIJELJENJE	DIV	$7 \text{ DIV } 2 = 3$
OSTATAK DIJELJENJA	MOD	$7 \text{ MOD } 2 = 1$

18

Relacijski operatori ...

- Uspoređivanje dvaju vrijednosti, rezultat je booleova vrijednost
 - Istina, Da, +, 1
 - Laž, Ne, -, 0

19

Relacijski operatori

VEĆE	>	$7 > 2 = \text{ISTINA}$
VEĆE ILI JEDNAKO	>=	$7 >= 2 = \text{ISTINA}$
MANJE	<	$7 < 2 = \text{LAŽ}$
MANJE ILI JEDNAKO	<=	$7 <= 2 = \text{LAŽ}$
JEDNAKO	=	$7 = 2 = \text{LAŽ}$
RAZLIČITO	<>	$7 <> 2 = \text{ISTINA}$

20

Logički operatori

- Argumenti logičkih izraza su boole-ove vrijednosti istina (1) i laž (0)
 - i (and) – konjunkcije
 - ili (or) – disjunkcije
 - ne (not) - negacije

21

Operator "i" $z = a \text{ and } b$

A	B	Z
LAŽ	LAŽ	LAŽ
LAŽ	ISTINA	LAŽ
ISTINA	LAŽ	LAŽ
ISTINA	ISTINA	ISTINA

22

Operator "ili" $z = a \text{ or } b$

A	B	Z
LAŽ	LAŽ	LAŽ
LAŽ	ISTINA	ISTINA
ISTINA	LAŽ	ISTINA
ISTINA	ISTINA	ISTINA

23

Operator "ne" $z = \text{not } a$

A	Z
LAŽ	ISTINA
ISTINA	LAŽ

24

Adresni operatori

- Rezultat je adresa memorijske lokacije varijable (@)
- Rezultat je vrijednost pohranjena u memorijskoj lokaciji na adresi na koju pokazuje kazalo (^)

25

Operatori niza

- Stapanje dvaju znakovnih nizova - +

26

Operatori usmjereni na rukovanje s bitovima ...

- BIT - najmanja informacijska jedinica, vrijednost 0 ili 1, pohrana cjelobrojnim tipom podatka (8 bita = 1 bajt)
- pomak bitova za određeni broj pozicija na lijevu stranu, desna strana se zamjenjuje s 0 - (shl - shift left)
- pomak bitova za određeni broj pozicija na desnu stranu, lijeva strana se zamjenjuje s 0 - (shr - shift right)

27

Operatori usmjereni na rukovanje s bitovima ...

- Izvođenje logičke operacije "i" nad odgovarajućim parom bitova
- Rezultantna vrijednost je 1 ako oba bita u paru imaju vrijednost 1, u protivnom rezultat je 0 - (and)
- Izvođenje logičke operacije "ili" (uključivo ili) nad odgovarajućim parom bitova.
- Rezultantna vrijednost je 0 ako oba bita u paru imaju vrijednost 0, u protivnom rezultat je 1 - (or)

28

Operatori usmjereni na rukovanje s bitovima

- Izvođenje logičke operacije "isključivo ili" nad odgovarajućim parom bitova.
- Rezultantna vrijednost je 1 ako su vrijednosti bitova u paru različite, u protivnom rezultat je 0 - (**xor**)
- Logičko komplementiranje, negacija bita, promjena 0 u 1 i 1 u 0 - (**not**)

29

Prioriteti primjene operatora

OPERATOR	KATEGORIJA	PRIORITET
@, NOT	UNARNI OPERATOR	NAJVIŠI
*, /, DIV, MOD, AND, SHL, SHR	BINARNI OPERATOR	
+, -, OR, XOR	BINARNI OPERATOR	
=, <>, <, >, <=, >=,	RELACIJSKI OPERATOR	NAJNIŽI

AKO POSTOJE, ZAGRADE UTVRĐUJU PRIORITETE PRIMJENE OPERATORA

Komentari

- Objašnjenja značenja varijabli, instrukcija, funkcija i napomene o programu - (*** ***)

31

Ulazno-izlazne operacije

- **writeln(var1, var2, ...);**
 - var = varijabla čija se vrijednost želi ispisati
 - vrijednosti varijabli se ispisuju prema zapisanom redoslijedu u jednom redu, a na kraju ispisa prelazi se na početak novog reda
- **write(var1, var2, ...);**
 - vrijednosti varijabli se ispisuju prema zapisanom redoslijedu u jednom redu, a na kraju ispisa ostaje se u istom redu
- **writeln(var1, ' ', var2, ' ', ...);**
 - bjeline između članova treba posebno navesti

32

Primjeri:

```
A := 1; B := 2; C := 3;
Predmet := 'racunalo';
Writeln(A, B, C);           123
Writeln(A, ' ', B, ' ', C); 1 2 3
Writeln('Malo', Predmet);
    Maloracunalo
Writeln('Malo ', Predmet, '.');  Malo
racunalo.
```

33

- Writeln(var:mjesta, ...);
 - mjesta = ukupni broj mjesta za ispis vrijednosti, vrijednosti se pozicioniraju udesno

34

Primjeri:

```
A := 10; B := 2; C := 100;
Writeln(A, B, C);           102100
Writeln(A:2,B:2,C:3);       10 2100
Writeln(A:3,B:2,C:3);       10 2100
Writeln(A,B:2,C:4);         10 2 100
X := 421.53;
Writeln(X); 4.2153000000E+02
Writeln(x:8); 4.21E+02
```

Realni brojevi se ispisuju u eksponencijalnom obliku

35

- Writeln(var:mjesta:decimale, ...);
 - decimale = broj mjesta za ispis decimalnog dijela vrijednosti realnog broja

36

Primjeri:

```
X := 421.53;
Writeln(X:6:2);           421.53
Writeln(X:8:2);           421.53
Writeln(X:8:4);           421.5300
```

37

Ulazna operacija

- Unos podataka s ulazne jedinice (tipkovnica, disk). Podaci se odvajaju bjelinama.
- Read(var1,var2, ...);
 - var =varijabla određenog tipa
 - pridruživanje vrijednosti varijablama u popisu
- Readln(var1,var2, ...);
 - pridruživanje vrijednosti varijablama u popisu s pomakom u novi red

38

UPRAVLJANJE IZVOĐENJEM PROGRAMA

39

Uvjetno izvođenje naredbi ...

- Skup naredbi u programu se izvodi samo ako je zadovoljen postavljeni uvjet
- Naredba ako (if) - naredba izbora


```
if izraz
then naredba1
else naredba2
```

 - **izraz** = booleov izraz – vrijednost izraza je istina (da) ili laž (ne)
 - ako je izraz istinit, izvodi se naredba1, u protivnom naredba2

40

Uvjetno izvođenje naredbi ...

```
if izraz
  then naredba1
```

– ako je **izraz** lažan, naredba1 se ne izvodi

41

Uvjetno izvođenje naredbi

```
if izraz
  then naredba
  else begin
    naredba_1;
    .....
    naredban
  end;
```

42

Složena naredba

```
begin
  naredba1, naredba2, ... naredban
end;
```

43

Naredba slučaja case – naredba izbora ...

```
case izbornik_slučaja of
  oznaka_slučaja1 : naredba1;
  oznaka_slučaja2 : naredba2;
  .....
  oznaka_slučajan : naredban;
end;
```

44

Naredba slučaja case – naredba izbora ...

- tip podatka **oznaka_slučaja** mora biti jednak tipu podatka **izbornik_slučaja**
- oznaka slučaja
 - jedna ili više konstanti
 - područje vrijednosti - dvije konstante odvojene znakom '..'

45

Naredba slučaja case – naredba izbora ...

- izvodi se naredba čija je oznaka slučaja jednaka tekućoj vrijednosti izbornika slučaja ili se nalazi unutar područja vrijednosti
- ako niti jedna oznaka slučaja ne sadrži vrijednost izbornika slučaja, ne izvodi se niti jedna naredba

46

Primjer

case Smjer of

```
'c', 'c7' : cest := cest + 1;
'v', 'v7' : vodni := vodni + 1;
'tt'      : telekom := telekom+ 1;
'pt', 'p7' : post := post + 1;
'st', 'z7' : zeljez := zeljez + 1;
'a', 'a7' : zracni := zracni + 1
```

end;

47

Naredbe petlje

- Izvođenje naredbi se ponavlja
 - While
 - For
 - Repeat

48

Naredba while

- **while** uvjet **do** naredba;
 - uvjet = booleov izraz
 - naredba = jednostavna ili složena naredba
- Izvođenje naredbe se ponavlja sve dok je uvjet zadovoljen
- Ako kod ulaza u naredbu **while** uvjet nije zadovoljen, naredba se niti jedanput ne izvodi

49

Naredba while - primjer

```
program Ponavljanje;
var
    A, B, Broj    : integer;
    Test          : boolean;
begin
    Write('Upisite dva broja: ');
    Readln(A, B);
    Test := A > B;
    Broj := 1;
    while Broj <= 5 do begin
        Writeln('A je veće od B ', Test);
        Broj := Broj + 1;
    end;
    Writeln('Svrsetak ispisa!')
end.
```

50

Naredba for ...

- **for** indeks := izraz1 **to** izraz2 **do** naredba;
 - indeks = indeksna varijabla skalarnog tipa (cijeli broj, realni broj, znak, booleova vrijednost)
 - izraz1 = početna vrijednost indeksne varijable
 - izraz2 = konačna vrijednost indeksne varijable
 - naredba = jednostavna ili složena naredba

51

Naredba for

- Broj izvođenja naredbe određen je vrijednošću indeksne varijable (indeks) koja se mijenja od početne (izraz1) do konačne (izraz2) vrijednosti s korakom jedan
 - to = povećanje indeksa za 1 (inkrementiranje)
 - downto = smanjenje indeksa za 1 (dekrementiranje)

52

Naredba for - primjer

Zadatak: Izračunati N faktoriijela

$F = N! = 1 * 2 * 3 * \dots * N$

$F := 1;$

for i := 1 **to** N **do**

$F := F * i;$

53

Naredba repeat

- **repeat** naredba **until** uvjet;
 - uvjet = booleov izraz
 - naredba = jednostavna ili složena naredba
- Izvođenje naredbe se ponavlja sve dok uvjet nije zadovoljen
- Naredba se izvodi barem jedanput jer se uvjet ispituje na kraju

54

Naredba repeat - primjer

```

program Dijeli;
var   A, B : integer;
      Kolicnik : real;
      Odgovor : char;
begin
  repeat
    Write('Upisite dva broja: ');
    Readln(A, B);
    Kolicnik := A / B;
    Writeln('Kolicnik: ', Kolicnik);
    Write('Zelite li zadati nove brojeve ? (D/N) ');
    Readln(Odgovor)
  until (Odgovor = 'n') or (Odgovor = 'N')
  end.
    
```

55

Zadatak

- Nacrtajte dijagram toka programa koji učitava varijablu pod nazivom ocjena za 1000 studenata. Potrebno je izračunati prosječnu vrijednost varijable ocjena. U programu je potrebno ne dozvoliti unos varijable koja je izvan područja vrijednosti 1-5.

56

Zadatak

- Nacrtajte dijagram toka programa koji učitava varijablu pod nazivom matični broj za 1000 studenata. Potrebno je izračunati broj studenata na vodnom smjeru ukoliko su njihovi matični brojevi u rasponu 3278-3311. Napomena: neki studenti su se ispisali tako da nisu svi brojevi dodijeljeni.

57

Potprogrami ...

- Primjena istog skupa naredbi na različitim skupovima podataka ili na različitim mjestima u programu
- Definiraju se na početku programa iza deklaracije varijabli
- Formalni parametri - navode se kod definicije potprograma
- Stvarni parametri - zamjenjuju formalne kod stvarnog računanja

58

Funkcije i procedure

- Funkcija - uz zadane vrijednosti argumenta vraća izračunatu vrijednost u program

$$X := \text{abs}(a);$$
- Procedura - se poziva kada se želi izvesti jedan ili više zadataka, bez vraćanja rezultata

$$\text{Writeln('ovo je pokus');}$$

59

Struktura definicije funkcije

```
function NazivFunkcije(formalni parametri) : tip rezultata;
const
    navođenje konstanti;
type
    definicija tipova podataka;
var
    navođenje varijabli;
    procedure i funkcije;
begin
    tijelo funkcije;
end;
```

60

Struktura definicije procedure

```

procedure NazivProcedure(formalni parametri);
const
    navođenje konstanti;
type
    definicija tipova podataka;
var
    navođenje varijabli;
    procedure i funkcije;
begin
    tijelo procedure;
end;
    
```

61

Struktura programa ...

```

program NazivPrograma;
const
    navođenje konstanti;
type
    definicija tipova podataka;
var
    navođenje varijabli;
    definicija procedura i funkcija;
begin
    glavno tijelo programa
end
    
```

62

Struktura programa

- Razlika strukture programa i potprograma:
 - ZAGLAVLJE (program, function, procedure)
 - Svršetak programa je točka, a potprograma točka zarez

63

Program za dijeljenje dva broja

```

program Dijeljenje;
var A, B : integer;
    Kolicnik : real;
(* Definicija procedure Upis *)
procedure Upis(var X, Y : integer);
(* X, Y Formalni parametri *)
(* var - Stvarni parametri moraju biti varijable - Poziv po imenu*)
begin
    Write('Zadajte dva broja: ');
    Readln(X, Y)
end;
function Dioba(I, J : integer) : real;
begin
    Dioba := I/J
end;
(* Poziv procedure Upis *)
Upis(A, B); (* A, B - Stvarni parametri *)
(* Poziv funkcije Dioba *)
(* Stvarni parametri ne moraju biti varijable - Poziv po vrijednosti*)
Kolicnik := Dioba(A, B);
Writeln('Kolicnik iznosi ', Kolicnik)
end.
    
```

64