

Sadržaj:

Uvodni sat	2
Dopunska nastava iz Računalstva	2
Općenito o predmetu	2
Potreban pribor	2
Kriterij ocjenjivanja	2
Pravila ponašanja	3
Osnovni pojmovi u programiranju	4
Programski jezik	5
Tko? Kada? Gdje? Zašto?	7
Podaci u programu (VARIJABLE i KONSTANTE)	12
Pretvorba tipova podataka	15
Pravila za automatsko pretvaranje tipova	15
Eksplicitna pretvorba	15
Vježbe – osnovni pojmovi u programiranju	19

Uvodni sat

Maja Carević

Dopunska nastava iz Računalstva

prof. Skuliber:

utorkom 13.15-14.00

srijedom 14.15-15.00

Općenito o predmetu

O Programskom jeziku C

- Zašto baš C?

<https://www.tiobe.com/tiobe-index/programming-languages-definition/>

<https://www.tiobe.com/tiobe-index/>

<http://www.moj-posao.net/Pretraga-Poslova/?keyword=&area=&category=11>

Potreban pribor

- Udžbenik – NOVI!!!
(Brođanac, Budin, Markučić, Perić: Uvod u računalstvo
Stranjak, Tomić: Udžbenik C jezik)
- Velika bilježnica „na kvadratiće“
- Pribor za pisanje i brisanje
- Računalo
 - o razvojna okolina za programski jezik C

<https://sourceforge.net/projects/orwelldevcpp/>

Kriterij ocjenjivanja

- Usmeno ispitivanje (6 pitanja):
 - o Ocjena nedovoljan – potpun odgovor na 0, 1 ili 2 pitanja
 - o Ocjena dovoljan – potpun odgovor na 3 pitanja
 - o Ocjena dobar – potpun odgovor na 4 pitanja
 - o Ocjena vrlo dobar – potpun odgovor na 5 pitanja
 - o Ocjena odličan – potpun odgovor na 6 pitanja
- Kratke pisane provjere, pisane provjere znanja:
 - o Ocjena nedovoljan <=49%
 - o Ocjena dovoljan 50%-59%
 - o Ocjena dobar 60%-74%

- o Ocjena vrlo dobar 75%-89%
- o Ocjena odličan 90%-100%

Pravila ponašanja

- Kućni red i Statut škole
 - o Dolaziti na nastavu na vrijeme
 - o Donošenje pribora
 - o Redovito pohađanje nastave i učenje
- Pravila ponašanja u praktikumu
 - o Pažljivo koristiti opremu tj. računala
 - o Zabranjeno korištenje računala u svrhe koje nisu za nastavu (internet, igrice, ...)
 - o Zabranjena instalacija programa
 - o Ugasiti računalo nakon vježbi

Osnovni pojmovi u programiranju

PROGRAM

- skup naredbi poredanih po strogo određenom pravilu nakon čijeg izvršavanja dobivamo određeni rezultat

NAREDBA ILI INSTRUKCIJA

- osnovni element programa

PROGRAMER

- osoba koje piše program

PROGRAMIRANJE

- proces nastajanja programa

KODIRANJE

- pisanje programa

PROGRAMSKI JEZIK

- umjetno stvoren jezik u kojima se pišu programi

Programski jezik

Programski jezik čini skup naredbi i pravila njihova pisanja.

Programski jezici su se razvijali paralelno s razvojem računala.

Podjela programskih jezika:

STROJNI JEZIK

Sve naredbe su zapisane u binarnom obliku

Jedini jezik koji računalo razumije

Naporan z aprogramera, mora pamtit i nizove 0 i 1

Teško se pronalaze greške

ASEMBLERSKI JEZIK

Jezik niže razine

Naredbe su označene kraticama koje podjećaju na svoju namjenu

ADD +, MUL *

VIŠI PROGRAMSKI JEZICI

Naredbe su kratke, lako pamtljive riječi

Programi postaju kraći i razumljiviji

BASIC, FORTRAN, Pascal, C/C++, C#, Java, Python

Odrednice programskog jezika:

LEKSIK- dopušten skup znakova

SINTAKSA- pravila pisanja naredbi

SEMANTIKA- značenje ispravno napisanih naredbi

ETAPE U PROGRAMIRANJU

1. ZADAVANJE ZADATKA

2. ANALIZA ZADATKA

3. SASTAVLJANJE ALGORITMA

ALGORITAM – razrađen postupak koji nas u konačnom broju koraka vodi prema rješenju problema

4. PISANJE PROGRAMA U ODABRANOM PROGRAMSKOM JEZIKU

kodiranje

5. PREVOĐENJE PROGRAMA

Budući da računalo razumije samo strojni jezik, programe pisane u višim programskim jezicima prevodimo u strojni jezik

PROGRAMI PREVODIOCI:

INTERPRETER- prevode naredbu po naredbu, ne stvaraju izvršni kod

KOMPAJLER- prevode čitav kod, stvaraju izvršnu verziju te ju izvršavaju

Otkrivaju se leksičke i sintaktičke pogreške

6. TESTIRANJE PROGRAMA

Provjerava se ispravnost programa

Za testiranje se uzimaju dobro odabrani ulazni podaci za koje znamo izlazne vrijednosti

Otkrivaju se logičke pogreške

7. IZRADA DOKUMENTACIJE

Objašnjava kako program radi (dokumentacija za programera) ili kako ga koristiti (korisnička dokumentacija)

8. ODRŽAVANJE PROGRAMA

Postupak mijenjanja programa tijekom njegovog životnog vijeka

Pronalaženje i ispravljanje pogreški, dodavanje nove funkcionalnosti

Tko? Kada? Gdje? Zašto?

1. Tko je osmislio programski jezik C?
2. Kada i gdje je osmišljen programski jezik C?
3. Zašto je osmišljen programski jezik C?
4. Što je ANSI C i čemu on služi?
5. Čemu služi programski jezik C?
6. Nabroji barem 4 primjene programskog jezika C.

Istraži:

1. Što je C++?
2. Tko je osmislio C++?
3. Čemu služi C++?

Programiranju u programskom jeziku C

Razvojna okolina (programski alat) koju koristimo:

Dev- C++ (<https://sourceforge.net/projects/orwelldevcpp/>)

- Osnovni sastavni dijelovi:
tekstualni editor za pisanje izvornog koda (kod programa)
kompajler
debugger

Rad u programskom alatu Dev- C++

Pisanje programskog koda	File->New->Source File	Ctrl+N
Spremanje programskog koda	File->Save	Ctrl+S
	File->Save As	
Otvaranje programskog koda	File->Open	Ctrl+O
Prevođenje programa (kompajliranje)	Execute->Compile	F9
Izvođenje programa	Execute->Run	F10
Kompajliranje i izvođenje programa	Execute->Compile&Run	F11

Ime datoteke u kojoj se nalazi kod (također i mapa u kojoj se datoteka s kodom nalazi) ne smije sadržavati razmake i točku te dijakritičke znakove (ćčžšđ), a ekstenzija je .c. Ime treba biti smisljeno.

(npr. Petlja_for1.c, if_1.c, helloworld.c)

Prvi program napisan u C-u:

Svaki program napisan u C-u treba poštovati strukturu programa tj. način pisanja programa.

helloworld.c

```
#include<stdio.h> //komentar
/*komentar
u više redova */
int main(void){
    //funkcija koja ispisuje hello world
    printf("Hello world!");
    return 0;
}
```

Naredba:	Objašnjenje:
<code>include</code>	Pretprocesorska naredba
<code>stdio.h</code>	Biblioteka funkcija
<code>main()</code>	Glavna i jedina obavezna funkcija u programu
<code>printf()</code>	Funkcija za ispis
<code>;</code>	Označava kraj jedne i početak druge naredbe
<code>/* */</code>	Komentar u više redova
<code>//</code>	Komentar u jednom retku
<code>{}</code>	Blok naredbi

Pretprocesorske naredbe

- počinju znakom #
- opći oblik naredbe: #naredba parametri
- nisu sastavni dio C-a
- povećavaju općenitost programa
- npr. `#include <ime_datoteke>` – kopira sadržaj navedene datoteke na mjesto naredbe `#include`

Biblioteke funkcija

- u C-u ne postoje ugrađene funkcije
- sve funkcije nalaze se u bibliotekama (dolaze zajedno sa kompajlerom)
- nastale su standardizacijom C-a
- na početku je potrebno navesti koju biblioteku ćemo upotrebljavati pa će u cijelom programu biti dostupne sve funkcije koje se nalaze u toj biblioteci
- npr.
 - `stdio.h`- (standard input output; standardni input je tipkovnica, a standardni output je ekran)-funkcije za ulaz i izlaz
 - `math.h`- matematičke funkcije
 - `string.h`- funkcije za rad sa znakovnim varijablama

Komentari

- `//` komentar u jednom retku
- `/* */` komentar zapisan u više redova

Funkcije za ispis

- Ispis teksta
 - `printf("niz znakova koje treba ispisati na zaslonu");`
 - Znakovne konstante

<code>\n</code>	Prelazak u novi red
<code>\t</code>	Horizontalni tabulator

2. Ispis vrijednosti

- Tipovi podataka u C-u:

cjelobrojni	int
realni	float
realni dvostruke preciznosti	double

znakovni	char
tip podataka koji ne sadrži vrijednost	void

- Formati za ispis:

Ispis cijelog broja	%d
Ispis realnog broja (decimalnog broja)	%f
Ispis realnog broja tipa double	%lf
Ispis jednog znaka	%c
Ispis niza znakova	%s

- Funkcija printf() ima oblik:
printf("lista formata", lista varijabli);
- Funkcija scanf() ima oblik:
scanf("lista formata", lista varijabli);
- Primjeri:

Naredba	Objašnjenje
printf("%d", 3);	Ispis broja 3
printf("%d", 3+2);	Ispis zbroja 2+3
printf("%d,%f",a,b)	Ispis vrijednosti cjelobrojne varijable a i realne varijable b
printf("%.2f",broj)	Ispis realne varijable broj zaokružene na dvije decimale
scanf("%d",&x);	Učitavanje cjelobrojne varijable x (& označava adresu varijable x)
scanf("%d,%f",&x,&y);	Učitavanje cjelobrojne varijable x i realne varijable y PAZI! Prilikom učitavanja potrebno je upisati zarez između vrijednosti varijabli
scanf("%d %f",&x,&y);	Učitavanje cjelobrojne varijable x i realne varijable y

Primjer: Program koji računa zbroj dva realna broja

```
#include<stdio.h>
int main(void){
    float a,b,c;
    printf("Ovo je program koji ispisuje zbroj dva realna broja\n");
    printf("Upisi dva realna broja:");
```

```
scanf("%f,%f",&a,&b);  
c=a+b;  
printf("Zbroj brojeva %.2f i %.2f iznosi %.2f\n.",a,b,c);  
return 0;  
}
```

Objašnjenje i uputa za pisanje programa:

- Uvijek treba naznačiti što program radi i dati uputu korisniku

```
printf("Ovo je program koji ispisuje zbroj dva realna broja\n");  
printf("Upisi dva realna broja:");
```

- Deklaracija varijabli

Ako u programu koristimo promjenjive vrijednosti, prije njihovog korištenja trebamo ih deklarirati. Deklaracijom se zadaju tip i ime varijable. Prilikom deklaracije u memoriji se rezervira određen memorijski prostor za varijablu ovisno o tipu.

Npr. 2 bajta za broj tipa int, 4 bajta za broj tipa float, ...

```
float a,b,c;
```

- Učitavanje vrijednosti varijabli a i b

```
scanf("%f,%f",&a,&b);
```

- Vrijednost zbroja varijabli a i b spremamo u novu varijablu c (pridružujemo varijabli c)

```
c=a+b;
```

- Ispis zbroja

```
printf("Zbroj brojeva %.2f i %.2f iznosi %.2f\n.",a,b,c);
```

Podaci u programu (VARIJABLE i KONSTANTE)

VARIJABLE - podaci čije se vrijednosti tijekom izvođenja programa mijenjaju

KONSTANTE - podaci čije se vrijednosti u programu ne mogu mijenjati

IDENTIFIKATORI - imena varijabli, konstanti i funkcija

- sastoje se od alfanumeričkih znakova (brojke i slova) i _
- prvi znak mora biti slovo ili donja crta (_)
- ne smiju biti ključne riječi
- C razlikuje mala i velika slova, $a \neq A$

KLJUČNE RIJEČI – identifikatori točno određenog značenja

- ključne riječi i sintaksa programskog jezika određuju što i kako možemo učiniti u određenom programskom jeziku
- 32 ključne riječi definirane standardom ANSI C
- pišu se malim slovima
- npr. *int, float, void*

Primjer: Zaokruži one identifikatore koji su ispravno napisani.

1) int, mjesec, a1, 1a, a 1, a_1, float

Rješenje:

1) mjesec, a1, a_1,

Primjer: Koliko različitih identifikatora je napisano?

Suma, suma, S, s, Suma

Rješenje: 5

KONSTANTE – vrijednosti koje se zadaju jedanput te se ne mijenjaju tijekom izvođenja programa

- cjelobrojne, realne, znakovne, konstanta niza znakova (string konstant)
- definiranje konstanti (preprocesorska naredba):
`#define ime_konstante vrijednost_konstante`
- npr. `#define pi 3.14`

Deklaracija i inicijalizacija varijabli

VARIJABLE - podaci čije se vrijednosti tijekom izvođenja programa mijenjaju

DEKLARACIJA – najava varijabli

- zadaje se tip i ime varijable
- opći oblik naredbe:
`tip_podataka lista_varijabli;`
- npr.
`int a, broj, x1, a_1;`
`float zbroj, z;`
- izvan svih funkcija, unutar funkcije, na početku bloka naredbi
- prilikom deklaracije u memoriji se rezervira memorijska lokacija za svaku varijablu navedenu u listi varijabli
veličina rezerviranog prostora ovisi o tipu varijable
(npr. 2 bajta za varijablu tipa int, 4 bajta za varijablu tipa float,...)

INICIJALIZACIJA – navođenje početne vrijednosti varijable

- prilikom deklaracije ili nakon nje, no prije prvog korištenja

Primjer:

- 1) Deklariraj cjelobrojnu varijablu broj
`int broj;`
- 2) Deklariraj i inicijaliziraj varijablu a na vrijednost 1.234
`float a=1.234;`
- 3) Definiraj konstantu e=2.71
`#define e 2.71`

Lokalne i globalne varijable

PODRUČJE DJELOVANJA VARIJABLE – ovisi o mjestu na kojem je deklarirana

LOKALNE VARIJABLE

– deklarirane na početku funkcije ili kao argument funkcije i mogu se koristiti samo u toj funkciji (ili bloku naredbi)

- nije definirana početna vrijednost

GLOBALNE VARIJABLE

– deklarirane izvan svih funkcija, mogu se koristiti u cijelom programu

- početna vrijednost je 0

Primjer:

```
#include<stdio.h>

int a,b=2;

float e,f=-1.2;

int main(void){
    int c=3,d;
    printf("a=%d,b=%d,c=%d,d=%d,e=%.2f,f=%.2f",a,b,c,d,e,f);
    return 0;
}
```

Lokalne varijable su:

Globalne varijable su:

Cjelobrojne varijable su:

Realne varijable su:

Deklarirane su varijable:

Inicijalizirane su varijable:

Što će se ispisati?

Rješenje:

Lokalne varijable su: c,d

Globalne varijable su: a,b,e,f

Cjelobrojne varijable su: a,b,c,d

Realne varijable su: e,f

Deklarirane su varijable: a,b,c,d,e,f

Inicijalizirane su varijable: b,f,c

Što će se ispisati? a=0, b=2, c=3, d=ne zna se, e=0, f=-1.20

Primjer: Lokalne i globalne varijable

```
#include<stdio.h>
int a,b=2;
float e,f=-1.2;
int main(void){
    int c=3,d;
    printf("a=%d,b=%d,c=%d,d=%d,e=%.2f,f=%.2f",a,b,c,d,e,f);
    return 0;
}
```

Lokalne varijable su:	
Globalne varijable su:	
Cjelobrojne varijable su:	
Realne varijable su:	
Deklarirane su varijable:	
Inicijalizirane su varijable:	
Što će se ispisati?	

DZ:

1. Napišite program u kojem ćete
 - deklarirati lokalnu cjelobrojnu varijablu *ocjena*
 - deklarirati globalnu znakovnu varijablu *slovo*
 - deklarirati globalne realne varijable *x, y*
 - definirat konstantu *g=9.81*
 - deklarirati i inicijalizirati lokalnu varijablu *a=2.44*
2. Zaokružite ispravno napisane identifikatore:
zbroj, X1, 1t, float, zbroj 1

Pretvorba tipova podataka

Dvije su vrste pretvorbe:

- automatska ili implicitna
- zadana ili eksplicitna

Pravila za automatsko pretvaranje tipova

1. najveću važnost imaju operandi tipa double pa float, long, int, short ili char
2. ako su svi operandi tipa int, tada je i rezultat tipa int
3. ako u izrazu postoji operand tipa double, svi podaci pretvaraju se u double

Pr:

```
int x=3;
float y=4.9;
```

x+y=?

Rj: x+y=3+4.9=7.9, float
/* Veću važnost ima operand tipa float pa je i rezultat tipa float*/

Pr:

```
int i,a;
float j=5.7,b=3.2;
i=j;
a=b;
i,a=?
```

Rj: i=5.7
/*pa će i biti 5 (odbacuju se decimale)*/
a=3.2
/*pa će a biti 3 (odbacuju se decimale)*/
Dakle, konačno rješenje i=5, a=3. Oba rezultata su tipa int.

Pr:

```
int i=5;
float j;
j=i;
j=?
```

Rj: j=5
/* j će biti 5.000000, to jest i se pretvara u realni broj te se pridružuje varijabli j */

Dakle, j=5.000000, float

Eksplicitna pretvorba

Cast operator

Opći oblik naredbe: (tip podataka) operand;

/* zagrade su oznake cast operatora*/

Pr: Ako su deklarirane i inicijalizirane varijable x, y i z odredi kojeg je tipa rezultat izraza te koji je rezultat izraza pod a), b), c) i d):

```
float x=5.6;
int y=2 ,z=3;
```

- a) $y/z+x$
- b) $(int) (x+5)$
- c) $(float)(y/z)+(int)(x)$
- d) $(float)(y)/z + (int)(3.26)$

Rj: Uvrštavamo:

a) $2/3+5.6=0+5.6=5.6$, float

b) $(int)(5.6+5)=(int)(10.6)=10$, int

/*odbacujemo znamenke*/

c)

$(float)(2/3)+(int)(5.6)=$

$=(float)(0)+5$

$=0.000000+5$

$=5.000000$, float

/*kad zbrajamo int i float rezultat će biti float*/

d)

$(float)(2)/3 + (int)(3.26)=$

$=2.000000/3+3=$

$=0.666667+3=$

$=3.666667$, float

/* prvo 2 pretvorimo u float, to je 2.000000, zatim dijelimo itd.*/

Zad: Ako su deklarirane i inicijalizirane varijable a, b i c odredi kojeg je tipa rezultat izraza te koji je rezultat izraza pod

a), b) i c):

```
float a=4.6;
int b=1 ,c=3;
```

- a) $a/b+c$
- b) $(int) (a+5)$

c) $(\text{int})(a) + ((\text{float})(b))/c$

Rj: sami riješite u bilježnicu!

Primjer:

Zapis algoritma korištenjem dijagrama toka: (na ploči)

Zapis algoritma u pseudojeziku:

```
{  
  Ulaz(a,b,c);  
  O=a+b+c;  
  S=O/2;  
  P=sqrt(s*(s-a)*(s-b)*(s-c));  
  Izlaz(O,P);  
}
```

Vježbe – osnovni pojmovi u programiranju

- 1) Napišite program pravokutnik.c koji računa opseg i površinu pravokutnika.
- 2) Napišite program krug.c koji računa opseg i površinu kruga.
Broj π definiraj kao konstantu.
pretprocesorska naredba: #define pi 3.1415
- 3) Napišite program koji računa volumen i oplošje kugle.

$$V = \frac{4}{3}r^3\pi, O = 4r^2\pi$$

Broj π definiraj kao konstantu.

- 4) Napišite program koji računa težinu tijela ako je zadana masa tijela ($G = mg, g = 9.81$).
 g definiraj kao konstantu
- 5) Napišite program koji računa udaljenost dviju točaka u koordinatnom sustavu.
$$A(x_1, y_1), B(x_2, y_2) d = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$$
- 6) Napišite program koji traži da se učita jedan troznamenkasti broj. Program ispisuje sve znamenke učitano broj.
- 7) Napiši program koji računa X za zadane A, B i C.
$$X = \frac{\sqrt{2A^3 - 3B}}{C}$$
- 8) Napiši program koji računa Z za zadane X i Y.
$$Z = \sqrt{\frac{X^5}{Y+4}}$$
- 9) Napišite program koji traži upisivanje jednog cijelog broja. Program ispisuje njegov prethodnik i sljedbenik.
- 10) Str. 54. u udžbeniku – zadaci koje nismo riješili

Relacijski i logički operatori

Relacijski operatori

-relacijski operatori rabe se kada je potrebno ustanoviti u kakvom su odnosu dvije vrijednosti

Operacija	Operator
manje od	<
veće od	>
jednako	==
manje ili jednako	<=
veće ili jednako	>=
različito	!=

-rezultati relacijskih izraza mogu biti samo cjelobrojne vrijednosti 0 (laž) i 1 (istina)

-relacijski operatori imaju niži prioritet od aritmetičkih

Logički operatori

Operacija	Operator
logička negacija(NOT)	!
logički i (AND)	&&
logički ili (OR)	

Prioritet	Operator
1.	!
2.	>,>=,<,<=
3.	==,!=
4.	&&
5.	

-rezultati logičkih izraza mogu biti samo 1 ili 0 (true ili false)

TABLICE ISTINITOSTI:

-konjunkcija i disjunkcija

A	B	A&&B	A B
0	0	0	0
0	1	0	1
1	0	0	1
1	1	1	1

-negacija

A	!A
1	0
0	1

Pr: Sljedeće tvrdnje zapiši pomoću relacijskih i logičkih operatora:

Rj:

a i b su negativni brojevi	$(a < 0) \&\& (b < 0)$
m je neparan broj	$m \% 2 == 1$ ili $m \% 2 != 0$
x je paran broj veći od 14	$(x \% 2 == 0) \&\& (x > 14)$
z je djeljiv sa 3 ili 5 ili 7	$(z \% 3 == 0) (z \% 5 == 0) (z \% 7 == 0)$
c je iz intervala $[10, 25 >$	$(c >= 10) \&\& (c < 25)$
d pri dijeljenju sa 7 ne daje ostatak 3	$d \% 7 != 3$

Uvjetno grananje

Jednostruko uvjetno grananje

<pre>if(uvjet) naredba;</pre>	<pre>if(uvjet){ blok_naredbi; }</pre>
-----------------------------------	---

Višestruko uvjetno grananje

<pre>if(uvjet) naredba1; else naredba2;</pre>	<pre>if(uvjet){ blok_naredbi1; } else{ blok_naredbi2; }</pre>
---	---

1. Program učitava broj a te ispisuje da li je broj djeljiv sa 5.
2. Program ispisuje da li učitani realni broj manji, veći ili jednak 0.
3. Program učitava 3 realna broja i ispisuje najveći među njima.
4. Program učitava broj a te ispisuje da li je broj paran ili je neparan.
5. Program koji učitava dva broja i od većeg učitano broj oduzima manji. Rezultat ispisuje na ekran.
6. Program koji provjerava da li je neki broj troznamenkast.
7. Program računa y za zadani x .

$$y = \begin{cases} -\frac{3}{2}x + 5, & x \geq 2 \\ 2x^3, & x < 2 \end{cases}$$

8. Program ispisuje apsolutnu vrijednost broja.

$$|x| = \begin{cases} x, & x \geq 0 \\ -x, & x < 0 \end{cases}$$

9. Program računa y za uneseni x .

$$y = \begin{cases} x^2, & x < 0 \\ 0, & x = 0 \\ \sqrt{x}, & x > 0 \end{cases}$$

10. Napiši program koji učitava cijeli broj manji od 10000. Program ispisuje poruku je li njegova znamenka desetice djeljiva s 3.
11. Napiši program koji provjerava da li je učitani troznamenkasti broj Armstrongov. Broj je Armstrongov ako je jednak zbroju kubova svojih znamenki.

Uvjetni operator (?)

`(uvjet)?(naredba_1):(naredba_2);`

`naredba_1` i `naredba_2` su aritmetički izrazi ili funkcije

Ako je uvjet ispunjen izvršava se `naredba_1`, a inače `naredba_2`.

zamjenjuje niz naredbi:

```
if(uvjet)
    naredba_1;
else
    naredba_2;
```

Primjer:

Koristeći uvjetni operator napiši naredbu koja varijablu `x` uvećava za 1 ako je `i > -2`, a inače `x` umanjuje za 3.

Primjer:

Program učitava dva broja i ispisuje veći broj.

Primjer:

Program ispisuje apsolutnu vrijednost broja bez korištenja funkcije `abs`.

$|x| = \begin{cases} x, & x \geq 0 \\ -x, & x < 0 \end{cases}$

Zadatak

Kolike će biti vrijednosti varijabli x,y i z nakon izvođenja naredbi:

a)

```
int x=2,y,z=4;
```

```
y=(x<z)?(z++):(x-z++);
```

b)

```
int x=9,y=3,z;
```

```
z=(x%y)?++x:(y+=x--);
```

Zadatak: (str. 60. 2.)

Koje vrijednosti će poprimiti varijable i,x i c nakon niza naredbi:

```
int i,c;
```

```
float d,x=5.2;
```

```
i=(int)(d+2.83);
```

```
(i%2)?x=(float)i+1:c=(int)(x+=15.3);
```

ako je vrijednost varijable d

a) 3.2

b) 5?

Naredba switch case

Opći oblik naredbe:

```
switch(izraz){  
    case konstanta_1: blok_naredbi_1;  
                      break;  
    case konstanta_2: blok_naredbi_2;  
                      break;  
    ...  
    case konstanta_n-1: blok_naredbi_n-1;  
                       break;  
    default: blok_naredbi_n;  
            break;  
}
```

Objašnjenje:

Najprije se izračuna vrijednost izraza, a zatim se uspoređuje s nizom konstanti (1 do n-1). Ako je dobivena vrijednost jednaka jednoj od konstanti, izvršava se blok naredbi pridružen toj konstanti. Ako dobivena vrijednost nije jednaka niti jednoj konstanti, izvršit će se blok naredbi koji pripada opciji default.

Izvršavanje bloka naredbi prekida se naredbom break (prouči primjer!).

Primjer 1:

Što će se ispisati ako je

- a) x=15
- b) x=9
- c) x=14

```
switch(x%7){  
    case 1: x+=5; break;  
    case 2: x-=2;  
    case 3: x*=3;  
    case 4: x+=2; break;  
    case 5: x-=10;  
    case 6: x*=4;  
    default: x+=3;  
}  
printf(„%d“,x);
```

Riješi zadatke koristeći naredbu switch-case:

Str 72. na dalje : Primjer 1, Primjer 4, Primjer 5

Str 75: Zadatak1

Zad: Program traži unošenje cijelog broja i u ovisnosti o ostatku pri dijeljenju učitano broj sa 7 ispisuje dan u tjednu (0 za ponedjeljak, 1 za utorak, ..., 6 za nedjelju).

Zad: Napravite program koji za zadane stranice pravokutnika računa opseg, površinu ili duljinu dijagonale zadanog pravokutnika.

$$P = ab, O = 2a + 2b, d = \sqrt{a^2 + b^2}$$

Zad: Napišite program, koristeći naredbu switch case, koji učitava cijeli broj i ispisuje pripadni mjesec u godini.

Zad: Napišite program koji učitava radijus kugle i ovisno o izboru korisnika računa oplošje ili volumen kugle.

Zadaci za ponavljanje – grananje

Zadaci za ocjenu dovoljan:

1. Program učitava broj a te ispisuje je li broj djeljiv sa 5. Ako je djeljiv ispisuje DA, a ako nije, ispisuje NE.
2. Program ispisuje da li je učitani realni broj jednak 0. Ako je jednak 0, ispisuje "broj je jednak 0".
3. Program ispisuje da li učitani realni broj manji, veći ili jednak 0.
4. Program učitava 3 realna broja i ispisuje najveći među njima.
5. Program učitava broj a te ispisuje da li je broj paran ili je neparan.
6. Program koji učitava dva broja i od većeg učitano broj oduzima manji. Rezultat ispisuje na standardni izlaz.
7. Program koji provjerava da li je neki broj troznamenkast.
8. Program ispisuje apsolutnu vrijednost broja.
9. Program ispisuje je li učitani broj djeljiv sa 10 i veći od 100
10. Napišite program koji učitava dva broja. Program ispisuje veći broj.
11. Program učitava tri realna broja i ispisuje da li je prvi broj veći od zbroja druga dva broja.
12. Program učitava cijeli broj a te ispisuje da li je učitani broj djeljiv s brojem 3.
13. Program učitava cijeli broj a te ispisuje da li je učitani broj veći od 100 ili manji.
14. Napravite program koji traži učitavanje realnih brojeva a i b . Program treba izračunati i ispisati na zaslonu vrijednost varijable c zadane formulom:

$$c = \begin{cases} -5a + 1, & a < b \\ 2a - 1, & a \geq b \end{cases}$$

Zadaci za ocjenu dobar:

1. Napiši program koji učitava cijeli broj manji od 10000. Program ispisuje poruku je li njegova znamenka desetice djeljiva s 3.
2. (Jesen 2014./2015.) Napišite program u pseudojeziku koji će učitati dvoznamenkasti broj D i ispisati je li zbroj njegovih znamenaka paran ili neparan. (Napomena: Ne treba provjeravati je li broj dvoznamenkast.)

Zadaci za ocjenu vrlo dobar i odličan:

1. Napiši program koji provjerava da li je učitani troznamenkasti broj Armstrongov. Broj je Armstrongov ako je jednak zbroju kubova svojih znamenki.
2. Napišite program koji učitava radijus kugle i ovisno o izboru korisnika računa oplošje ili volumen kugle.
3. Zadaci u kojima koristimo uvjetni operator

Zadaci s državne mature DM:

4. Na naturalnoj večeri ukupno je D djevojaka i M mladića i svi žele zaplesati. Pleše se u parovima (jedna djevojka i jedan mladić). Budući da broj djevojaka i mladića nije jednak, oni koji nemaju svojega para stajat će sa strane. Napišite program u pseudojeziku koji će učitavati broj djevojaka D i mladića M , a zatim će ispisati koliko učenika neće plesati.
5. Neka je T broj koji se dobiva iz dvoznamenkastoga broja D tako da se zamijene mjesta znamenaka jedinice i desetice. Napišite program u pseudojeziku koji će učitati dvoznamenkasti prirodan broj D , a ispisati veći od brojeva D i T . (Napomena: Ne treba provjeravati je li broj dvoznamenkast.)
6. Marija se tijekom ljeta zaposlila u restoranu gdje priprema sendviče. U jedan sendvič osim peciva stavljaju se 4 kriške šunke i 3 kriške sira. Mariju je dočekalo M kriška šunke i N kriška sira i zanima ju koliko će najviše sendviča uspjeti napraviti od postojećih sastojaka. Napišite program u pseudojeziku koji će učitati prirodan broj M (broj kriška šunke) i prirodan broj N (broj kriška sira), a ispisati će najveći mogući broj sendviča koji se mogu napraviti od

postojećih sastojaka.

Napomena: Pretpostavite da ima dovoljno peciva.

Primjer ulaza

Ulaz Izlaz Objašnjenje

53

60

13 Šunke ima za 13 sendviča, a sira ima za 20 sendviča.

Dakle, moguće je napraviti ukupno 13 sendviča s obama sastojcima.

Zadaci za vježbu – petlja for

1. Potrebno je ispisati brojeve od 1 do 20. Ispis neka bude oblika:

Ispis brojeva od 1 do 20:

1 2 3 4 5 6 7 8 920

2. Treba ispisati parne brojeve u rasponu od 50 do 100. Ispis neka bude oblika:

Parni brojevi iz intervala od 50 do 100 su:

50 52 54 56..... 98 100

3. Treba ispisati parne brojeve u rasponu od 100 do 50. Ispis neka bude oblika:

Parni brojevi iz intervala od 100 do 50 su:

100 98 96 94 52 50

4. Potrebno je zbrojiti prvih N prirodnih brojeva. Ispis neka bude oblika:

Upisi broj do kojeg zelis zbrajati:

Zbroj prvih ... prirodnih brojeva je

5. Potrebno je prebrojiti sve prirodne brojeve djeljive sa 7 iz raspona od M do N. Ispis neka bude oblika:

Upisi pocetnu vrijednost raspona:

Upisi završnu vrijednost raspona:

U rasponu od ... do ... ima ... brojeva djeljivih sa 7.

6. Program prebrojava sve prirodne brojeve djeljive sa 7 iz raspona od M do N.

Upisi pocetnu vrijednost raspona: 1

Upisi završnu vrijednost raspona: 100

U rasponu od 1 do 100 ima 14 brojeva djeljivih sa 7.

7. Potrebno je ispisati djelitelje odabranog prirodnog broja. Po unosu broja treba provjeriti da li je uneseni broj prirodan, ako nije valja ponoviti unos. Ispis neka bude oblika:

Upisi prirodni broj:

Djelitelji broja ... su: ...

8. Potrebno je ispisati sve troznamenkaste brojeve čiji je zbroj znamenaka 5, a zadnja im je znamenka 0 (znamenka jedinice). Ispis neka bude oblika:

Troznamenkasti brojevi čiji je zbroj znamenaka 5, a zadnja im je znamenka 0 su:

...

...

Zadaci za ponavljanje – programske petlje

Za dovoljan:

1. Program ispisuje sve brojeve od 1 do 100 (korištenjem for, while i do-while petlje).
2. Program ispisuje sve brojeve od 100 do 1 (korištenjem for, while i do-while petlje).
3. Program ispisuje sve brojeve od m do n (korištenjem for, while i do-while petlje). M i n unosimo preko tipkovnice.
4. Program učitava 5 brojeva te ispisuje zbroj brojeva većih od 0, te ukupan broj negativnih brojeva (korištenjem for, while i do-while petlje).
5. Program učitava brojeve dok ne upišemo broj 100. Program ispisuje koliko brojeva je djeljivo s 10.
6. Program učitava brojeve dok ne upišemo broj 0. Program ispisuje zbroj brojeva koji su djeljivi sa 3.
7. Program učitava brojeve dok ne upišemo broj 0. Program ispisuje koliko brojeva smo unijeli.
8. Program učitava prirodni broj n. Program ispisuje sve prirodne brojeve u obrnutom poretку, od n do 1.
9. Napišite program koji traži da se učitaju 5 brojeva. Program ispisuje zbroj negativnih brojeva (korištenjem for, while i do-while petlje).
10. Napišite program koji ispisuje sve brojeve koji su djeljivi sa 10 između 50 i 150 te njihov zbroj.
11. Napišite program koji ispisuje sve brojeve u padajućem poretку između 100 i 1 koji su djeljivi sa 2. Također, program ispisuje zbroj takvih brojeva.
12. Napišite program koji traži da se učitaju četiri broja. Program ispisuje ukupan broj brojeva većih od -3.
13. Napišite program koji korisniku omogućava učitavanje brojeva sve dok se ne unese broj 0 ili broj veći od 200. Program ispisuje zbroj brojeva djeljivih sa 10.
14. Napišite program u C-u koji će učitavati cijele brojeve dok se ne unese 100 pozitivnih brojeva. Program treba na kraju ispisati zbroj svih učitanih brojeva.

Za dobar:

1. Program ispisuje zbroj parnih znamenki učitanih cijelog broja.
2. Program ispisuje koliko znamenki učitanih broja je veće od 5.
3. Program traži da se učitaju brojevi n. Program ispisuje kvadratne korijene prirodnih brojeva do n.
4. Napišite program koji traži da se učitavaju brojevi sve dok se ne unese broj 3 ili broj 5. Program ispisuje sve brojeve koji su veći od 15.

Za vrlo dobar i odličan:

1. Program ispisuje sumu svih četveroznamenkastih brojeva koji su djeljivi sa 17, a znamenka stotice im je veća ili jednaka 5.
2. Program ispisuje aritmetičku sredinu prvih

Problemski zadaci:

36. Skupina od N učenika pristupila je provjeri znanja iz Informatike. Napišite program u C-u koji će učitati broj učenika N i ocjenu za svakoga učenika te će ispisati srednju ocjenu cijele skupine.

36. Dva igrača igraju igru u kojoj naizmjenice stavljaju kamenčiće na niz polja. Prvi igrač stavlja jedan kamenčić na prvo polje i nakon toga drugi igrač stavlja dva kamenčića na drugo polje. U svakome sljedećem koraku igrač koji je na redu stavlja na sljedeće polje dva puta više kamenčića nego što je stavljeno na prethodno polje. Igra završava kada je broj kamenčića koje je stavio neki igrač u posljednjemu koraku veći ili jednak zadanomu broju N . Napišite program u pseudojeziku koji će učitati prirodan broj N , a ispisati u kojemu će koraku K završiti igra.

36. Hotelski lanac nabavlja ribu. Želi kupiti 100 kg ribe. Napišite program u pseudojeziku koji će unositi masu pojedine ribe sve dok ukupna masa ne prijeđe 100 kg. Program treba ispisati koliko je ukupno komada riba kupio hotelski lanac.

Petlja u petlji:

1. Napišite program koji ispisuje koliko brojeva između 10 i 200 ima zbroj znamenki djeljiv sa 2.
2. Ispiši sve Armstrongove brojeve između 50 i 1000.