

1.

Funkcije

TEORIJSKI ZADATCI

1.1. Definiranje i deklaracija funkcije

1. Napiši opći oblik funkcije u programskom jeziku C. Ukratko objasni značenje svakog dijela funkcije.

2. Objasni značenje sljedećih gotovih funkcija u C-u i pored njih napiši u koju biblioteku pripadaju.

Funkcija	Biblioteka	Značenje
<code>double sqrt(double x);</code>		
<code>int isdigit(int c);</code>		
<code>double fabs(double x);</code>		
<code>int isupper(int c);</code>		
<code>double pow(double x, double y);</code>		

3. Napiši primjer jednog programa u kojem ćeš pozvati barem tri funkcije iz 2. zadatka. Rješenje testiraj s pomoću prevoditelja programskog jezika C.

4. Napiši kako općenito glasi prototip funkcije u programskom jeziku C. Napiši i prototip funkcije kojom se provjerava je li neki znak u stringu slovo. Odredi kojeg je tipa ta funkcija, kako se zove te koje parametre sadržava.

5. Kojeg tipa mogu biti funkcije? Objasni, što funkcija vraća kao rezultat, ovisno o tome kojeg je tipa.

_____ - _____
_____ - _____
_____ - _____
_____ - _____

6. U zaglavlju funkcije pišemo _____ parametre. Oni se povezuju sa _____ parametrima iz glavnog programa.

7. Parametri se iz glavnog programa u funkciju mogu prenositi na dva načina (napiši engleske nazive):

_____ i _____.

Objasni razliku između dva navedena prijenosa.

8. Napiši zaglavlje te pokaži kako općenito pišemo glavnu funkciju `main`. Kojeg je tipa, od kojih se parametara sastoji te što joj je povratna vrijednost?

9. _____ varijable su definirane unutar tijela funkcije i njihove vrijednosti nisu vidljive u ostalim dijelovima programa.

10. _____ varijable su definirane u glavnoj funkciji `main` te se nalaze izvan ostalih napisanih funkcija.

11. Zadana je funkcija naziva `opseg`.

Odgovori na pitanja:

a) Kako se zovu parametri koji su navedeni u zaglavlju funkcije? _____

b) Ispiši lokalne varijable iz zadane funkcije. _____

```
float opseg(float x, float y){
    float o;
    o = 2*(x + y);
    return o;
}
```

12. Navedi tri razloga zašto se koristimo funkcijama prilikom pisanja složenijih programa.

13. Zaokruži slova ispred točno napisanih općih oblika programa koji sadržavaju funkciju kojoj je zaglavlje (prototip) `int funkcija(int x)`.

a)

```
#include <stdio.h>

int funkcija(int x) {
    //tijelo_funkcije;
    return rez;
}

int main(void){
    ...
    x = funkcija(a);
    ...
    return 0;
}
```

b)

```
#include <stdio.h>

int main(void){
    int funkcija(int x) {
        //tijelo_funkcije;
        return rez;
    }
    ...
    x = funkcija(a);
    ...
    return 0;
}
```

c)

```
#include <stdio.h>
int funkcija(int x);

int main(void){
    ...
    x = funkcija(a);
    ...
    return 0;
}

int funkcija(int x) {
    //tijelo_funkcije;
    return rez;
}
```

d)

```
#include <stdio.h>

int main(void){
    ...
    x = funkcija(a);
    ...
    return 0;
}

int funkcija(int x) {
    //tijelo_funkcije;
    return rez;
}
```

14. Napiši prototipove sljedećih funkcija:

a) funkcijom `Povrsina` računa se površina kvadrata

b) funkcija `Korijen` provjerava je li korijen nekog prirodnog broja i dalje prirodan broj; ako jest funkcija vraća `TRUE`, a u suprotnom `FALSE`.

c) funkcijom `Ars` računa se prosjek triju cijelih brojeva

d) funkcija `Trokut` vraća u glavni program opseg trokuta; parametri funkcije su tri stranice koje su racionalni brojevi

e) funkcija `Broji` računa koliko se jedinica nalazi u troznamenkastom binarnom broju.

1.2. Poziv funkcije

1. Kako se funkcija poziva u glavni program?

2. Ako smatraš da je tvrdnja točna, zaokruži slovo T. U suprotnom zaokruži slovo N.

a) Broj stvarnih i formalnih parametara ne treba biti jednak.

T	N
---	---

b) Formalni i stvarni parametri mogu imati isto ime.

T	N
---	---

3. Odredi redoslijed kojim se izvršava program kada dolazi do poziva funkcije. U prazne kućice upiši brojeve od 1 do 3.

Funkcija vraća rezultat na mjesto gdje je bila i pozvana.

Stvarni parametri se kopiraju u formalne parametre (funkcija dobiva ulazne podatke).

U funkciji se izvršava niz naredbi.

4. Odredi kako će izgledati ispis na zaslonu računala u sljedećim primjerima:

a)

```
#include <stdio.h>

int funkc1(int a){
    return 2 * a;
}

int main(void){
    printf("%d", funkc1(5));
    return 0;
}
```

Ispis:

b)

```
#include <stdio.h>

float funkc2(float a){
    float rez = 1;
    rez += a;
    return rez;
}

int main(void){
    printf("%.2f", funkc2(2.5));
    return 0;
}
```

Ispis:

c)

```
#include <stdio.h>

int funkc3(int x);

int main(void){
    printf("%d\n", funkc3(2));
    printf("%d", funkc3(-2));
    return 0;
}

int funkc3(int x){
    if(x < 0)
        return -x;
    return x;
}
```

Ispis:

d)

```
#include <stdio.h>

int funkc4(int x, int y);

int main(void){
    printf("%d\n", funkc4(3, 4));
    printf("%d\n", funkc4(4, 4));
    printf("%d\n", funkc4(4, 3));
    return 0;
}

int funkc4(int x, int y){
    if(x < y)
        return 1;
    else
        return 0;
}
```

Ispis:

5. Opiši riječima što rade funkcije `Funk1` i `Funk2` te za vrijednost varijable `n = 145`, odredi što će se ispisati na zaslonu računala. Što radi glavni program?

Postupak:

```
#include <stdio.h>

int Funk1(int a) {
    int s = 0;
    do{
        s = s + a % 10;
        a = a / 10;
    }while(a > 0);
    return s;
}

int Funk2(int x) {
    int br = 0;
    do{
        br++;
        x = x / 10;
    }while(x > 0);
    return br;
}

int main(void) {
    int n;
    float rez;
    printf("Upisi broj:\n");
    scanf("%d", &n);

    rez = float(Funk1(n))/Funk2(n);
    printf("rez = %4.2f\n", rez);

    return 0;
}
```

6. Riješi zadatke vezane za program iz 5. zadatka.

- a) Ispiši sve formalne parametre. _____
- b) Ispiši sve stvarne parametre. _____
- c) Ispiši sve lokalne varijable. _____
- d) Ispiši sve globalne varijable. _____

7. Odredi koju će vrijednost vratiti funkcija i kako će izgledati ispis u glavnom programu. Poziv funkcije u glavnom programu dan je unutar naredbe za ispis:

```
printf("%5.2f", fj(2, 3, 4));
```

```
float fj(int a, int b, int c) {
    c = b++;
    a = --c;
    c--;
    b /= 2;
    b = 10 * c % 3;
    c = pow(c, 2);
    a += sqrt(abs(2 * b - c));
    return a+c;
}
```

8. Zadano je zaglavlje funkcije (prototip) sa sljedećim parametrima:

_____ Horizontalna(float Fg, int l, int h);

- a) U zaglavlju funkcije navedeni su _____ parametri.
- b) Kojeg tipa mora biti funkcija ako vraća realnu vrijednost? Nadopiši na praznu crtu.
- c) Napiši cijelu funkciju Horizontalna tako da funkcija vraća vrijednost sile F definirane formulom.

Formula: $F = F_g \frac{h}{l}$ (F_g – sila teža, l – duljina kosine, h – visina kosine)

- d) Navedene su naredbe kojima se poziva funkcija Horizontalna. Ako je naredba napisana točno, stavi kvačicu, a u suprotnom ispravi pogreške. Objasni pogreške riječima.

1. horsila = horizotnalna(120, 5, 2);

2. horsila = Horizontalna(120.0, 5, 2, 7);

3. horsila = Horizontalna(127.5, 4, 6);

- e) Zadan je i glavni dio programa. Nadopuni prazne linije koda tako da program ispravno poziva funkciju Horizontalna i koristeći se njome računa iznos horizontalne sile.

```
int main(void){
    float sila, horsila;
    int dulj, vis;

    _____ //u varijablu sila učitati silu težu
    _____ //u varijablu dulj učitati duljinu kosine
    _____ //u varijablu vis učitati visinu kosine

    horsila = _____ //poziv funkcije
    printf("Horizontalna sila je: %5.3f",horsila);
    return 0;
}
```

- f) Kako će izgledati ispis na zaslonu računala ako su zadane vrijednosti: sila = 120, dulj = 5 i vis = 2?

9. Zadan je prototip funkcije sa sljedećim parametrima:

```
float Tlak(float visina, int gustoca, float konst);
```

Označi pogreške i ispravi naredbe kojima se poziva ta funkcija u glavnom programu ako su pozivi pogrešno napisani. Ako je naredba napisana točno, stavi kvačicu.

- a) rez = Tlak(2, 3500, 10);
- b) rez = tlak(2.0, 3500, 10);
- c) rez = Tlak(2.0; 3500; 10);
- d) rez = Tlak(2, 3500, 10.0):

10. Nadopuni prazne linije koda tako da funkcija Hipotenuza računa vrijednost hipotenuze pravokutnog trokuta i vraća njezinu vrijednost u glavni program.

```
#include <stdio.h>
#include <math.h>

_____ Hipotenuza (float x, float y){
    float c;
    c = _____; //izracun c
    return c;
}
int main(void){
    float a, b, pom;
    scanf(_____); //unos kateta

    pom = _____; //poziv f
    printf("%4.2f", pom);
    return 0;
}
```

11. Zadana je funkcija:

```
_____ Funk(int a, int i) {
    int j, p = 1;
    for (j = 1; j <= i; j++){
        p = p * a;
    }
    return p;
}
```

- a) Nadopiši na praznu crtu kojeg je tipa funkcija ako znamo da funkcija vraća cjelobrojnu vrijednost.
- b) Odredi koliko parametara ima funkcija. Navedi ih. Kako nazivamo te parametre?
- _____
- c) Ima li ova funkcija lokalne varijable? Ako da, koje su?
- _____

d) Što će se ispisati u glavnom programu nakon naredbe:

```
printf("Vrijednost je: %4d",Funk(3, 4));
```

e) Što će se ispisati u glavnom programu nakon naredbe:

```
printf("Vrijednost je: %4.2f",Funk(2.5, 2));
```

f) Što će se ispisati u glavnom programu nakon naredbe:

```
printf("Vrijednost je: %4.2f",Funk(2, 2, 3));
```

g) Napiši zadanu funkciju koristeći se do-while petljom.

h) Napiši matematički izraz koji opisuje ovu funkciju.

i) Nadopuni prazne crte u glavnom dijelu programa u kojem ćeš pozvati zadanu funkciju. Kako nazivamo varijable kojima se koristimo u glavnom dijelu programa?

```
int main(void){
    int x, y, w;

    //u varijablu x učitati prvi prirodan broj
    _____

    //u varijablu y učitati drugi prirodan broj
    _____

    w = _____ //poziv funkcije Funk
    printf("%d", w);

    return 0;
}
```