

Laboratorijska vježba 7

Cilj ove vježbe je upoznavanje sa konceptom funkcije u programskom jeziku C. Nakon vježbe student bi trebao/la biti u stanju pisati jednostavne funkcije. U uvodnom dijelu vježbe, studenti se upoznaju sa pravilima pisanja čitljivog koda koja će se primjenjivati ostatak semestra.

I UVODNI DIO VJEŽBE

Unesite slijedeći program:

```
#include <stdio.h>
void crtaj (int duzina)
{
    int i;
    for (i=1; i<=duzina; i++) {
        printf("*");
    }
}
int main()
{
    int i, niz[5];

    /* Unos pet vrijednosti */
    printf("\n Unesite vrijednosti za grafikon\n");
    for (i=0; i<5; i++) {
        printf("%d. vrijednost: ", i+1);
        scanf("%d", &niz[i]);
    }

    printf("\n      ^\n      |");
    for (i=0; i<5; i++) {
        printf("\n      |");
        crtaj(niz[i]);
        printf(" %d", niz[i]);
        printf("\n      |");
    }
    printf("----->\n");
    return 0;
}
```

- Analizirajte program.
- Kompajlirajte program.
- Testirajte program tako što ćete tri puta unositi različite ulazne vrijednosti. Prepravite program tako da se grafikon iscrtava koristeći neke druge znakove osim minusa i zvjezdica.

II ZADACI ZA SAMOSTALNU VJEŽBU

2. Napravite funkciju "faktorijel" koja računa faktorijel datog broja. Zatim iskoristite ovu funkciju za program koji računa sumu:

$$S(x) = \sum_{i=1}^n \frac{x}{i!}$$

Primjer ulaza i izlaza:

```
Unesite broj n u intervalu [1, 12]: 1
Unesite realan broj x: 2.45
Suma od 1 do 1 za x = 2.450 je 2.450
```

3. Napravite funkciju

```
int prost(int x)
```

koja vraća 1 (logička istina) ako je broj x prost, a 0 (logička neistina) ako x nije prost. Zatim iskoristite ovu funkciju u programu koji ispisuje sve proste brojeve između 1 i 100 (svaki broj u zasebnom redu).

4. Napišite funkciju "obrnut" koja vraća primljeni pozitivan cijeli broj okrenut naopako. Npr. ako se funkcija pozove sa vrijednošću 12345 (dvanaest hiljada tristo četrdeset i pet), treba vratiti broj 54321 (pedeset četiri hiljade tristo dvadeset i jedan). Pri tome nije dozvoljeno koristiti nizove niti raditi bilo kakav ispis u funkciji! Napravite i kraći program koji testira tu funkciju.

5. Sinus broja x može se izračunati koristeći Taylorov razvoj:

$$\sin(x) = \sum_{i=1}^n (-1)^{i-1} \frac{x^{2i-1}}{(2i-1)!}$$

Potrebno je napraviti funkciju "sinus" koja na ovaj način računa sinus. Pored broja x čiji se sinus računa, ova funkcija treba imati i parametar n koji određuje preciznost rješenja: veće n daje tačniju vrijednost sinusa, no obratite pažnju da faktorijel koji se nalazi u nazivniku vrlo brzo raste tako da će kod dovoljno velikog n biti prekoračen opseg vrijednosti tipa int.

Pri rješavanju zadatka iskoristite funkciju **pow** iz biblioteke **math.h**, te funkciju **faktorijel** koju smo napravili u zadatku 2. Zatim napravite program koji poredi tačnost ove funkcije i ugrađene funkcije **sin(x)** (u biblioteci **math.h**) za neke vrijednosti x i n unesene sa tastature.

Primjer ulaza i izlaza:

```
Unesite x: 1
Unesite n: 4
sin(x)=0.841471
```

```
sinus(x)=0.841468
Razlika: 0.000003 (0.00%).
```

6. Napraviti program koji ispisuje na ekranu Pascalov trougao sa **n** redova pri čemu se prirodan broj **n** unosi sa tastature:

```
      1
     1 1
    1 2 1
   1 3 3 1
  1 4 6 4 1...
```

Svaki broj u Pascalovom trouglu je jednak sumi dva broja koji se nalaze iznad njega. Iznad je crvenom bojom označen broj 4 koji je jednak zbiru brojeva 1 i 3. Za računanje broja na koordinatama x,y napisati i koristiti funkciju `pascal(int x, int y)`. Tako broj 4 iz prethodnog primjera ima koordinate x=5, y=2.

Primjer ulaza i izlaza programa:

```
Unesite n: 10
```

```
1
1  1
1  2  1
1  3  3  1
1  4  6  4  1
1  5 10 10 5  1
1  6 15 20 15 6  1
1  7 21 35 35 21 7  1
1  8 28 56 70 56 28 8  1
1  9 36 84 126 126 84 36 9  1
```

Savjet za rješavanje: Da bismo lakše riješili ovaj zadatak, predstavimo Pascalov trougao kao trougaonu matricu (kao na primjeru ulaza i izlaza datom iznad).

Svaki član ove matrice možemo označiti kao $P(x,y)$ pa tako je $P(1,1)=1$, $P(3,2)=2$, $P(5,3)=6$ itd. Kada ovako formulišemo problem, elemente $P(x,y)$ možemo izračunati na dva načina: preko binomnog obrasca ili rekursivno.

Binomni obrazac daje koeficijente za rastavljanje n -tog stepena binoma (sume oblika $x+y$):

$$(x + y)^n = \sum_{k=0}^n \binom{n}{k} x^k$$

Binomni obrazac se izračunava po formuli:

$$P(m, n) = \binom{m}{n} = \frac{m!}{n!(m-n)!}$$

Pogledajmo *rekurzivno* rješenje zadatka (funkcija koja poziva samu sebe): Pascalov trougao se može opisati sljedećom formulom:

$$P(m, n) = \begin{cases} 1, & \text{ako } m = 1 \text{ ili } n = 1 \\ P(m-1, n) + P(m-1, n-1), & \text{ako } m > n > 1 \end{cases}$$