



UNIVERZITET U NOVOM SADU
FAKULTET TEHNIČKIH NAUKA
KATEDRA ZA PRIMENJENE RAČUNARSKE NAUKE

Osnovi programiranja i programskih jezika

Računarske vežbe – vežba 7

Zimski semestar 2025/2026.

Studijski program: Informacioni inženjering

Funkcije, prenos parametara Datoteke

Funkcije

- Potprogrami u C-u se realizuju preko **funkcija**
- Omogućuju **dekompoziciju problema** na manje, lakše rešive probleme
- Korišćenjem funkcija se omogućuju **ponovno korišćenje koda**

Funkcije

- Funkcije **komuniciraju sa ostalim delovima programa preko:**
 - **ulaznih i/ili ulazno/izlaznih parametara**
 - **povratne vrednosti**
- **Sve promenljive deklarisanе u funkciji su vidljive samo unutar te funkcije**

Funkcija za sabiranje dva broja (primer)

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

```
/* Definicija funkcije */
```

```
int zbir(int a, int b) {
```

```
    int c;
```

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

```
    return c;
```

```
    /* Ovo je krace moglo da bude napisano kao return a+b; */
```

```
}
```

```
int main() {
```

```
    int c;
```

```
    /* Poziv funkcije */
```

```
    c = zbir(3,5);
```

```
    printf("%d\n", c);
```

```
    /* Ovo smo krace mogli da napisemo kao */
```

```
    printf("%d\n", zbir(3,5));
```

```
    return 0;
```

```
}
```

Funkcija za sabiranje dva broja – II način

```
#include <stdio.h>

/* Deklaracija funkcije */
int zbir(int, int);

int main() {
    /* Poziv funkcije */
    printf("%d\n", zbir(3,5));
    return 0;
}

/* Definicija funkcije */
int zbir(int a, int b) {
    return a+b;
}
```

Primer 2

```
#include <stdio.h>
float power(float x, int k);

int main() {
    /* Poziv funkcije */
    float s = power(2.0,8);
    printf("%f\n", s);
    return 0;
}

/* stepenuje x^k tako sto k puta pomnozi x */
float power(float x, int k) {
    int i;
    float rezultat = 1;
    for (i = 0; i < k; i++) rezultat *= x;
    return rezultat;
}
```

Naredba `return`

- Ova naredba omogućava funkciji da vrati neku vrednost, i istovremeno se izvršavanje funkcije završava
- Vrednost koja se vraća mora odgovarati tipu koji je specificiran kao povratni tip funkcije
- Funkcija može sadržavati više `return` naredbi, ali prva na koju se naiđe završava izvršavanje funkcije i vraća navedenu vrednost

Parametri funkcije

- Svaki parametar deklarira lokalnu promenljivu koja je vidljiva samo u telu funkcije
- Vrednost ove promenljive se postavlja na vrednost koja je saopštena prilikom poziva funkcije (vrednost prosleđenog argumenta)
- Prestaje da postoji neposredno nakon izlaska iz funkcije, tj. kad se izvrši **return**
- Važni koncepti:
 - Lokalna promenljiva u funkciji je potpuno različita od promenljive koja je prosleđena prilikom poziva funkcije
 - Ukoliko funkcija izvrši promenu vrednosti jednog od svojih parametara, to ne utiče na promenljivu čija je vrednost prosleđena funkciji

Prenos parametara

```
#include <stdio.h>
// PO VREDNOSTI
void f(int i) {
    i = 3;
}

int main() {
    int i = 5;
    f(i);
    printf("%i", i);
    return 0;
}
```

```
#include <stdio.h>
// PO REFERENCI
void f(int *i) {
    *i = 3;
}

int main() {
    int i = 5;
    f(&i);
    printf("%i", i);
    return 0;
}
```

Zadatak I.

Nacrtati dijagram toka algoritma i napisati C program kojim se učitavaju dužine kateta i računa dužina hipotenuze. Za računanje hipotenuze napisati funkciju hipotenuza.

Primer 3 – ispis niza

```
void ispis(int *a, int n) {  
    int i; // lokalna promenljiva  
  
    printf("[");  
    for (i = 0; i < n; i++){  
        printf("%d", a[i]);  
        if (i != (n-1)) printf(", ");  
    }  
    printf("]");  
}
```

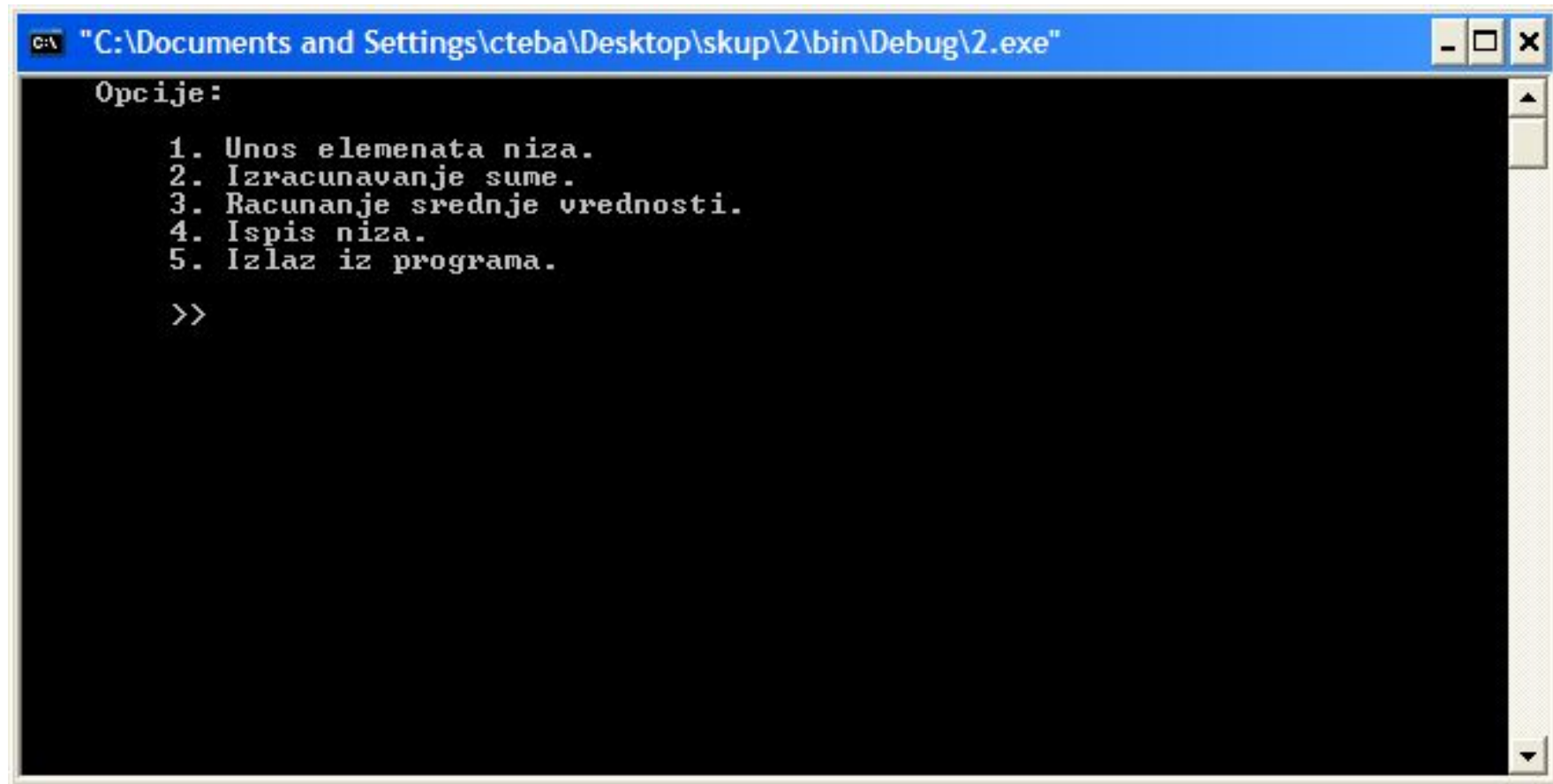
Zadatak 2.

Nacrtati dijagram toka algoritma i napisati C program kojim se učitava niz od n elemenata, a zatim omogućiti korisniku da bira neku od sledećih radnji:

- izračunavanje sume elemenata niza,
- računanje srednje vrednosti elemenata,
- nalaženje minimuma,
- nalaženje maksimuma.

Omogućiti izvršavanje više radnji (jedna za drugom).

Zadatak 2 – izlaz na konzoli



The screenshot shows a Windows command prompt window with a blue title bar. The title bar text is "C:\Documents and Settings\cteba\Desktop\skup\2\bin\Debug\2.exe". The window content is black with white text. The text displayed is:

```
Opcije:  
  1. Unos elemenata niza.  
  2. Izracunavanje sume.  
  3. Racunanje srednje vrednosti.  
  4. Ispis niza.  
  5. Izlaz iz programa.  
  
>>
```

Zadatak 2 - main

```
4 void unosNiza(int *, int *);
5 int suma(int *, int);
6 double avg(int *, int);
7 void ispisi(int *, int);
8
9 int main() {
10
11     int a[MAX_SIZE];
12     int n, opcija;
13
14     do {
15         printf("    Opcije:\n\n");
16         printf("\t1. Unos elemenata niza.\n");
17         printf("\t2. Izracunavanje sume.\n");
18         printf("\t3. Racunanje srednje vrednosti.\n");
19         printf("\t4. Ispis niza.\n");
20         printf("\t5. Izlaz iz programa.\n ");
21         printf("\n\t>> ");
22         scanf("%d", &opcija);
23
24         switch( opcija ) {
25             case 1: unosNiza(a, &n); break;
26             case 2: printf("Suma je: %d.\n\n", suma(a, n)); break;
27             case 3: printf("Srednja vrednost je: %lf.\n\n", avg(a, n)); break;
28             case 4: ispisi(a, n); break;
29         }
30
31     } while ( opcija != 5);
32
33     return 0;
34 }
```

Zadatak 2 - main

```
void unosNiza(int *, int *);
int suma(int *, int);
double avg(int *, int);
void ispis(int *, int);

int main() {
    int a[MAX_SIZE], n, opcija;

    do {
        printf("## Opcije:\n\n");
        printf("\t1. Unos elemenata niza.\n");
        printf("\t2. Izracunavanje sume.\n");
        printf("\t3. Racunanje srednje vrednosti.\n");
        printf("\t4. Ispis niza.\n");
        printf("\t5. Izlaz iz programa.\n");
        printf("\n\t>> ");
        scanf("%d", &opcija);

        switch (opcija) {
            case 1: unosNiza(a, &n); break;
            case 2: printf("Suma je: %d. \n\n", suma(a, n)); break;
            case 3: printf("Srednja vrednost je: %lf.\n\n", avg(a, n)); break;
            case 4: ispis(a, n); break;
        }
    } while (opcija != 5);

    return 0;
}
```

Zadatak za domaći I.

Dati su prirodni brojevi n , m ($n \leq m$). Nacrtati dijagram toka algoritma i napisati C program kojim se određuje koji od brojeva od n do m predstavljaju prestupne godine. Godina je prestupna ako je zadovoljen sledeći skup uslova:

1. broj godine je deljiv sa četiri, i
2. važi jedno od sledećih pravila:
 - broj godine nije deljiv sa 100
 - broj godine je deljiv sa 400

Zadatak 3.

Nacrtati dijagram toka algoritma i napisati C program kojim se štampaju svi trocifreni brojevi (ako ih ima) koji su jednaki sumi faktoriijela svojih cifara.

Zadatak za domaći 2.

Nacrtati dijagram toka algoritma i napisati C program koji učitava prirodan broj n , a zatim koristeći funkciju **prost** štampa sve proste brojeve manje od datog broja n .

Datoteke

Datoteke služe za **dugotrajno čuvanje podataka**,
smeštene su u masovnoj memoriji

Tipovi datoteka u C-u:

- **tekstualne**
- **binarne**

Datoteke

- Deklarisanje promenljive tipa datoteke

```
FILE *pf;
```

- Otvaranje datoteke

```
FILE *fopen(const char *nd, const char *r);
```

- Režimi za tekstualne datoteke
 - 'r' - čitanje
 - 'w' - pisanje
 - 'a' - dodavanje

Datoteke

- Čitanje/pisanje podataka iz/u tekstualnu datoteku datoteku
 - `fscanf/fprintf` - formatirani ulaz/izlaz
 - `fgets/fputs` - ulaz/izlaz za tekst
- Zatvaranje datoteke

```
int fclose(FILE *pf);
```

Primer 4 - čitanje

```
#include <stdio.h>
#define MAXL 80

int main() {
    FILE *pf;
    char str[MAXL];

    pf = fopen("test.txt", "r");

    if(pf != NULL) {
        while(fgets(str, MAXL, pf) != NULL)
            puts(str);
        fclose(pf);
    } else {
        printf("Nije moguće otvoriti datoteku ili datoteka ne postoji.");
    }

    return 0;
}
```

Zadatak 4.

Nacrtati dijagram toka algoritma i napisati C program za prebrojavanje reči iz ulazne datoteke. U izlaznoj datoteci ispisati broj reči i najdužu pronađenu reč.

Zadatak 5.

Nacrtati dijagram toka algoritma i napisati C program kojim se iz ulazne datoteke učitava niz studenata koji su položili ispit (indeks, ime, ocena).

- U izlaznu datoteku "najbolji.txt" ispisati podatke o studentu sa najvećom ocenom
- U izlaznu datoteku "najlosiji.txt" ispisati podatke o studentu sa najnižom ocenom
- Na standardnom izlazu ispisati prosečnu ocenu svih studenata

Zadatak za domaći 3.

- Napisati funkciju:

```
void kodiraj(char *ul_dat, char *izl_dat)
```

koja kodira ulaznu datoteku tako što svako slovo iz ulazne datoteke pomera za jednu poziciju unapred, u alfabetu. Parametri `ul_dat` i `izl_dat` su nazivi ulazne, odn. izlazne datoteke.

- Napisati funkciju:

```
void dekodiraj(char *ul_dat, char *izl_dat)
```

koja dekodira ulaznu datoteku tako što svako slovo iz ulazne datoteke pomera za jednu poziciju unazad, u alfabetu. Parametri `ul_dat` i `izl_dat` su nazivi ulazne, odn. izlazne datoteke.

- Napisati program koji testira prethodno definisane funkcije.