

## Prvi kolokvijum iz Programiranja 2

### Napomene:

- a) Pažljivo proučite Uputstvo pre popunjavanja Obrasca za odgovore.  
 b) Vrednost odgovora: tačan = **15**; netačan = **-3.75**; nevažeći (nula ili više zacrtnjenih kružića) = **0**.  
 c) Na pitanjima se može osvojiti najviše **45** poena.  
 d) Zadatak nosi **55** poena.

### I ZADACI

#### Zadatak

Napisati program na programskom jeziku C koji vrši obradu nad zadatom sekvencom dezoksiribonukleinske kiseline (DNK). DNK sekvenca se sastoji od dva lanca komplementarnih nukleotida: adenina (A – 0), guanina (G – 1), timina (T – 2) i citozina (C – 3). Jednoslovne oznake i kodovi nukleotida su dati u zagradama. Potrebno je napisati program koji proverava da li je uneta DNK sekvenca korektna. DNK sekvenca je korektna ukoliko se na korespondentnim mestima u oba DNK lanca nalaze parovi A – T i G – C, odnosno postoji komplementarnost nukleotida. DNK sekvenca se predstavlja pomoću dva niza celih brojeva jednake dužine koji predstavljaju DNK lance. Napisati program na programskom jeziku C koji se sastoji od sledećih stavki:

1. Program najpre treba da učitava dužinu DNK sekvence ( $0 < N < 1000$ ), a zatim i same elemente dva niza celih brojeva koji predstavljaju DNK lance od kojih se sekvenca sastoji.
2. Program treba da izvrši proveru da li se u zadatoj sekvenci nalaze samo dozvoljeni kodovi nukleotida (0, 1, 2, 3). Ukoliko se u sekvenci nalaze dozvoljeni kodovi, program ispisuje DOZVOLJENI KODOVI, u suprotnom ispisuje NEDOZVOLJENI KODOVI i završava izvršavanje. Poruka se ispisuje nakon kompletne provere.
3. Program treba da ispiše učitane nizove u zasebnom redu korišćenjem odgovarajućih slovnih oznaka za nukleotide razdvojenih crticom.
4. Program treba da izvrši proveru da li je DNK sekvenca zadata u vidu dva DNK lanca korektna. Sekvenca je korektna ukoliko su nukleotidi na korespondentnim mestima u oba DNK lanca komplementarni. Ukoliko je sekvenca komplementarna ispisuje SEKVENCA JE KOMPLEMENTARNA, u suprotnom ispisuje SEKVENCA NIJE KOMPLEMENTARNA.

Primer ulaza:	Primer izlaza:
10 3 1 0 0 2 0 2 3 3 1 1 3 2 2 0 2 0 1 1 3	DOZVOLJENI KODOVI C-G-A-A-T-A-T-C-C-G G-C-T-T-A-T-A-G-G-C SEKVENCA JE KOMPLEMENTARNA
5 3 1 4 0 0 1 3 2 2 2	NEDOZVOLJENI KODOVI

#### Zadatak

Napisati program na programskom jeziku C koji vrši obradu nad zadatom sekvencom ribonukleinske kiseline (RNK). RNK sekvenca se sastoji od jednog lanca nukleotida: adenina (A – 0), guanina (G – 1), timina (T – 2), citozina (C – 3) i uracila (U – 4). Jednoslovne oznake i kodovi nukleotida su dati u zagradama. Smatrati da postoji komplementarnost nukleotida A – T i G – C. Sekvenca dezoksiribonukleinske kiseline (DNK) se sastoji od dva RNK lanca koji sadrže samo komplementarne nukleotide A – T i G – C na korespondentnim mestima u oba RNK lanca koji čine DNK. Potrebno je napisati program koji na osnovu unete RNK sekvence formira odgovarajuću DNK sekvencu, ukoliko je to moguće. Smatrati da se jedna RNK sekvenca predstavlja pomoću niza celih brojeva, dok se DNK sekvenca predstavlja pomoću dva niza celih brojeva jednake dužine koji predstavljaju DNK lance. Napisati program na programskom jeziku C koji se sastoji od sledećih stavki:

1. Program najpre treba da učitava dužinu RNK sekvence ( $0 < N < 1000$ ), a zatim i same elemente niza celih brojeva koji predstavljaju RNK lanac.

2. Program treba da izvrši proveru da li se u zadatoj sekvenci nalaze samo dozvoljeni kodovi nukleotida (0, 1, 2, 3, 4) . Ukoliko se u sekvenci nalaze dozvoljeni kodovi, program ispisuje DOZVOLJENI KODOVI, u suprotnom ispisuje NEDOZVOLJENI KODOVI i završava izvršavanje. Poruka se ispisuje nakon kompletne provere.
3. Program treba da ispiše učitane nizove u zasebnom redu korišćenjem odgovarajućih slovnih oznaka za nukleotide razdvojenih crticom.
4. Program treba da formira odgovarajuću DNK sekvencu komplementiranjem zadate RNK sekvence. Ukoliko je RNK sekvencu moguće komplementirati ispisuje komplementarnu sekvencu, u suprotnom ispisuje KOMPLEMENTARNA SEKVENCA NEMOGUCA.

Primer ulaza:	Primer izlaza:
10 3 1 0 0 2 0 2 3 3 1	DOZVOLJENI KODOVI C-G-A-A-T-A-T-C-C-G G-C-T-T-A-T-A-G-G-C
5 3 1 4 0 0	DOZVOLJENI KODOVI C-G-U-A-A KOMPLEMENTARNA SEKVENCA NEMOGUCA

## **Zadatak**

Napisati program na programskom jeziku C koji vrši obradu nad zadatim sekvencama dezoksiribonukleinske kiseline (DNK). Jedna DNK sekvencu se sastoji od dva lanca komplementarnih nukleotida: adenina (A – 0), guanina (G – 1), timina (T – 2) i citozina (C – 3). Jednoslovne oznake i kodovi nukleotida su dati u zagradama. Potrebno je napisati program koji učitava dve DNK sekvence i proverava da li su unete DNK sekvence korektne. Zatim, program treba da proveriti koliko su dve unete DNK sekvence slične. Jedna DNK sekvencu se predstavlja pomoću dva niza celih brojeva jednake dužine koji predstavljaju DNK lance. Napisati program na programskom jeziku C koji se sastoji od sledećih stavki:

1. Program najpre treba da učitava dužinu DNK sekvenci ( $0 < N < 1000$ ), a zatim i same elemente četiri niza celih brojeva koji predstavljaju DNK lance.
2. Program treba da izvrši proveru da li se u zadatim sekvencama nalaze samo dozvoljeni kodovi nukleotida (0, 1, 2, 3) . Ukoliko se u sekvenci nalaze dozvoljeni kodovi, program ispisuje DOZVOLJENI KODOVI, u suprotnom ispisuje NEDOZVOLJENI KODOVI i završava izvršavanje. Poruka se ispisuje nakon kompletne provere.
3. Program treba da ispiše učitane nizove u zasebnom redu korišćenjem odgovarajućih slovnih oznaka za nukleotide razdvojenih crticom.
4. Program treba da utvrdi stepen sličnosti između dve unete DNK sekvence. Sličnost se određuje kao celobrojni procentualni udeo broja jednakih parova na korespondentnim mestima u obe sekvence u lancu u odnosu na ukupan broj parova (dužinu sekvence).

Primer ulaza:	Primer izlaza:
10 3 1 0 0 2 0 2 3 3 1 1 3 2 2 0 2 0 1 1 3 1 1 0 3 2 0 0 2 3 1 3 3 2 1 0 2 2 0 1 3	DOZVOLJENI KODOVI C-G-A-A-T-A-T-C-C-G G-C-T-T-A-T-A-G-G-C G-G-A-C-T-A-A-T-C-G C-C-T-G-A-T-T-A-G-C 80%
5 3 1 4 0 0 1 3 2 2 2	NEDOZVOLJENI KODOVI

## **Kraj**

## **II PITANJA**

### **Pitanje**

Na nekom računaru A realni brojevi se predstavljaju na širini od 10 bita u formatu **seeeemmmmm**, gde je **s** predznak broja, **eeee** eksponent u kodu sa viškom 7 i **mmmmmm** su bitovi normalizovane mantise. Na nekom računaru B realni brojevi se predstavljaju na širini od 12 bita u formatu **seeeemmmmmmm**, gde je **s** predznak broja, **eeeeee** eksponent u kodu sa viškom 15 i **mmmmmm** su bitovi normalizovane mantise. Mantisa na oba računara je u obliku  $1 \leq M < 2$ . Ukoliko se na oba računara izvrši smeštanje broja -49.283, koliko iznosi apsolutna vrednost razlike vrednosti broja smeštenog na računaru A u odnosu na vrednost broja smeštenog na računaru B? Sva zaokruživanja se obavljaju prema pravilima ANSI/IEEE standarda za realne brojeve.

(R) 0.25<sub>10</sub>

(R) 0<sub>10</sub>

(R) 0.5<sub>10</sub>

### Pitanje

Na jednom računaru, realni brojevi se predstavljaju na širini od 10 bita u formatu **seeeemmmmm**, gde je **s** bit predviđen za kodiranje predznaka broja, **eeee** su 4 bita za eksponent u kodu sa viškom 8, a **mmmmmm** su bitovi normalizovane mantise sa skrivenim bitom ( $0.5 \leq M < 1$ ). Celi brojevi na ovom računaru se predstavljaju u drugom komplementu na širini od 7 bita. Vrednost realnog broja na lokaciji X je 42.5<sub>10</sub>. Predstava celog broja na lokaciji Y je 123<sub>8</sub>. Koji je sadržaj realne promenljive Z nakon što se na ovom računaru obavi operacija  $Z = X + Y$ ? Sva zaokruživanja se obavljaju prema pravilima ANSI/IEEE standarda za realne brojeve.

(R) 350<sub>16</sub>

(R) 351<sub>16</sub>

(R) 360<sub>16</sub>

### Pitanje

Šta ispisuje sledeći program na programskom jeziku C?

```
#include <stdio.h>
void main() {
    int i = 0;
    while (2020) {
        switch (i++ * 2) {
            case 0: i != i;
            case 1: printf("%d ", i); continue;
            case 2: if (i % 2) break;
            default: printf("%d ", ++i);
        }
        if (i % 3) break;
    }
}
```

(R) 0 3 5

(R) 1 3 5

(R) 0 3 6

### Pitanje

Šta ispisuje sledeći program na programskom jeziku C?

```
#include <stdio.h>
void main() {
    int i = 3;
    while (2020) {
        switch (i-- * 2) {
            case 0: i != i;
            case 1: printf("%d ", i); continue;
            case 2: if (i % 2) break;
            default: printf("%d ", --i);
        }
        if (i % 3 == 0) break;
    }
}
```

(R) 0 -1 -3

(R) 1 -1 -3

(R) 2 0 -1

### Pitanje

Šta ispisuje sledeći program na programskom jeziku C?

```
#include <stdio.h>
void main() {
    int i = 0;
    while (2020) {
        switch (i++ * 2) {
            case 0: i != i;
            case 1: printf("%d ", i++); continue;
            case 2: if (i % 2) break;
            default: printf("%d ", i);
        }
        if (i % 3) break;
    }
}
```

®)0 3 4

(®) 1 3 4

®)0 4 5

### **Pitanje**

Šta ispisuje sledeći program na programskom jeziku C?

```
#include <stdio.h>
int main(void) {
    unsigned char arr[] = { 0x18, 10, 016 };
    //unsigned char arr[] = { 030, 0xb, 15 };
    //unsigned char arr[] = { 25, 014, 0x13 };
    unsigned char neg, lso, left, right;
    int n = sizeof(arr) / sizeof(unsigned char);
    for(int i = 0; i < n; i++) {
        neg = ~arr[i] + 1;
        lso = arr[i] & neg;
        left = arr[i] + lso;
        right = ~left & arr[i];
        right /= (2 * lso);
        printf("%d ", left | right);
    }
    return 0;
}
```

®)32 12 16

(®) 33 12 19

®)25 11 15