

## Programiranje 1 – ispit, 2. 9. 2024.

**Napomene:** Svako rješenje napišite isključivo na papir sa zadatkom, jer jedino njega predajete. Pomoćne račune smijete raditi na drugim papirima koje će vam dati dežurni asistent. U svim zadacima **obavezno** pišite postupak! Dozvoljeno je korištenje isključivo pribora za pisanje i brisanje, te službenog podsjetnika. Kalkulatori, te razne tablice, papiri i sl. nisu dozvoljeni! **Mobitele** isključite i pospremite daleko od sebe! Ako se ustanovi da **kod sebe** imate mobitel za vrijeme ispita, ispit se poništava i pokreće se stegovni postupak protiv vas.

**Zadatak 1** ( $4+3+4+4=15$  bodova)

- (a) Neka je  $b \geq 2$  prirodan broj i neka su  $x$  i  $y$  nenul znamenke u bazi  $b$ . Dokažite da je

$$(xxx)_{(b)} + (xx)_{(b)} \neq (yyy)_{(b)}.$$

- (b) Izračunajte bez pretvaranja u bazu 10:  $(AAA)_{(12)} + (13)_{(12)} \cdot (11)_{(12)}$ .

- (c) Neka su  $a, b$  i  $c$  logičke varijable koje se pojavljuju u logičkom izrazu  $f$ . Izraz  $f$  je istinit ako i samo ako je binarni broj  $(abc)_{(2)}$  prost (ako je varijabla istinita, pripadna znamenka je 1, a inače je 0). Odredite tablicu istinitosti te konjunktivnu **ili** disjunktivnu normalnu formu izraza  $f$ .

- (d) U nekom tekstnom dokumentu zapisan je niz rečenica. Zapis je formatiran tako da je unutar zagrada navedeno nekoliko rečenica, a rečenice su međusobno odvojene zarezom i razmakom. Svaka rečenica se sastoji od jedne ili više riječi odvojenih razmakom. Riječi se sastoje od malih slova engleske abecede, s izuzetkom prve riječi u rečenici koja počinje velikim slovom. Na kraju svake rečenice je točka, upitnik ili uskličnik. Zadnja rečenica sastoji se od samo jedne riječi.

Primjer zapisa jednog niza je sljedeći:

(Bok\_mama.,\_Volim\_te\_mama!,\_Das\_mi\_pet\_kuna?,\_Pliz.)

Napišite regularan izraz koji prepoznaje ovako zapisan niz rečenica.

## Programiranje 1 – ispit, 2. 9. 2024.

### Zadatak 2 (5+5=10 bodova)

- (a) Objasnite što je enumeracija. Napišite definiciju enumeracije koja reprezentira dane u tjednu. Napišite deklaraciju varijable tipa definirane enumeracije i postavite joj vrijednost na konstantu Ponedjeljak.
- (b) Napišite C-program koji implementira jednostavan kalkulator koristeći `if-else` naredbu. Kalkulator prima dva realna broja, te znak koji reprezentira operaciju ('+' - plus, '-' - minus, '\*' - puta i '/' - podijeljeno). Kalkulator računa rezultat operacije nad ulaznim brojevima te ga ispisuje u komandni prozor.

## Programiranje 1 – ispit, 2. 9. 2024.

### Zadatak 3 (6+19=25 bodova)

Neka je  $p_0(x) = a_0$  konstantni polinom. Rekurzivno definiramo niz polinoma  $p_1(x), p_2(x), p_3(x), \dots$  tako da je

$$p_n(x) = \sum_{k=0}^n p_{n-1}(k)^2 x^k \text{ za } n \geq 1.$$

- (a) Napišite pomoćnu funkciju `double eval(int k, double a[], double y)` koja prima nenegativan cijeli broj  $k$ , niz  $a_0, a_1, \dots, a_k$  i realan broj  $y$  te vraća  $\sum_{j=0}^k a_j y^j$  koristeći Hornerov algoritam. Ako ne koristite Hornerov algoritam, možete dobiti maksimalno 1 bod na ovom podzadatku.
- (b) Napišite funkciju `double mega_eval(double a_0, double y)` koja prima realne brojeve  $a_0$  i  $y$ . Funkcija treba vratiti vrijednost  $p_{20}(y)$ . Kad god evalirate bilo koji polinom u nekoj točki, treba koristiti funkciju `eval` iz (a) podzadatka. Smijete koristiti tu funkciju čak i ako je niste napisali. Smijete koristiti pomoćna polja.

**NAPOMENA:** U ovom zadatku smijete koristiti **najviše** tri polja duljine 21

## Programiranje 1 – ispit, 2. 9. 2024.

### Zadatak 4 (25 bodova)

Za prirodan broj  $d$  veći od 1 i prirodan broj  $a$ , definiramo  $v_d(a)$  kao najveći nenegativan cijeli broj  $m$  takav da je  $d^m$  djelitelj od  $a$ .

Napišite funkciju `void divisible_sort(int n, int a[], int d, double *avg)` koja prima prirodan broj  $n$ , niz različitih prirodnih brojeva  $a_0, a_1, \dots, a_{n-1}$  i prirodan broj  $d$  veći od 1.

Funkcija treba sortirati niz  $a_0, a_1, \dots, a_{n-1}$  tako da za  $i < j$  vrijedi da je  $v_d(a_i) < v_d(a_j)$  ili  $v_d(a_i) = v_d(a_j)$  i  $a_i < a_j$ . Preko varijabilnog argumenta `avg` treba vratiti prosječnu vrijednost od  $v_d(a_i)$  u nizu.

Treba napisati i program koji korisniku omogućava da učitava niz brojeva  $a_0, a_1, \dots, a_{n-1}$ , sortira ga koristeći napisanu funkciju za  $d = 5$ , te ispisuje sortirani niz i prosječnu vrijednost od  $v_5(a_i)$ .

## Programiranje 1 – ispit, 2. 9. 2024.

**Zadatak 5** (5+ 20 + [5] = 25 + [5] bodova)

- (a) Napišite funkciju `void kanta(int a[], int n, int b[])` koja prima niz  $(a[0], a[1], \dots, a[n-1])$  koji se sastoji od nenegativnih cijelih brojeva manjih od 10 i njegovu duljinu  $n$ , te niz cijelih brojeva  $(b[0], b[1], \dots, b[9])$ . Funkcija treba u  $b[i]$  upisati broj pojava broja  $i$  u nizu  $(a[0], a[1], \dots, a[n-1])$ .
- (b) Napišite funkciju `void histogram(int a[], int n, int *h, int *argh)` koja prima niz  $(a[0], a[1], \dots, a[n-1])$  koji se sastoji od nenegativnih cijelih brojeva manjih od 10 i njegovu duljinu  $n$ . Funkcija treba ispisati sveukupno 10 stupaca sastavljenih od simbola '\*', tako da se  $i$ -ti stupac sastoji od onoliko zvjezdica koliko puta se broj  $i-1$  pojavljuje u zadanom nizu. Stupci međusobno trebaju biti odvojeni razmakom. Možete pretpostaviti da će se svaki broj od 0 do 9 pojaviti barem jednom u nizu.

Na primjer, za niz  $(1, 1, 2, 3, 2, 9, 0, 9, 1, 4, 4, 5, 6, 7, 7, 7, 8)$ , ispis treba izgledati ovako:

```
      *           *
     * *      *   *   *
    * * * * * * * * * *
```

Nadalje, funkcija treba u varijabilne argumente `h` i `argh` zapisati visinu najvišeg stupca histograma i neki broj koji se pojavljuje najviše puta u nizu.

Smijete koristiti funkciju iz (a) podzadatka.

- (c) (Dodatnih 5 bodova koji se ne ubrajaju u uvjet od 80 posto) Napišite funkciju `void sortiranje_kantom(int a[], int n)` koja prima niz  $(a[0], a[1], \dots, a[n-1])$  koji se sastoji od nenegativnih cijelih brojeva manjih od 10 i njegovu duljinu  $n$ , te sortira niz. Smijete koristiti funkciju iz (a) podzadatka. Podzadatak treba riješiti u linearnoj složenosti  $\mathcal{O}(n)$ , u protivnom nećete dobiti bodove na njemu.