

## PRAKTIKUM IZ PROGRAMIRANJA 2

### - domaći zadatak broj 1 -

Sastaviti program na programskom jeziku C koji vrši određenu vrstu obrade nad ulaznim nizom/nizovima. Program treba da:

- a) učitava sve potrebne podatke (dužinu niza, elemente niza, itd.);
- b) izvrši zahtevanu obradu;
- c) ispiše sve dobijene rezultate po tačno definisanom formatu u tekstu zadatka;

Zavisno od rednog broja problema, svaki student treba da sastavi jedan od programa koji su dati u prilogu ovog dokumenta.

Prilikom rešavanja zadatka, predvideti maksimalni broj elemenata niza (najviše 1000) pri deklaraciji niza, a u toku rada proveravati da li je uneta dimenzija niza takva da svi elementi koje treba uneti mogu biti smešteni u niz predviđenih dimenzija. Ukoliko dužina niza nije korektna, potrebno je korektno prekinuti program i ne ispisivati ništa. Tip elemenata niza odabrati prema potrebama zadatka, odnosno koristiti proizvoljni tip tamo gde to nije suštinski bitno za sam algoritam.

Radi boljeg testiranja programa, odabrati nekoliko skupova podataka sa kojima će program biti testiran. Svaki primer treba da sadrži ulazne podatke i očekivani izlaz za te podatke.

#### **Napomene:**

1. Rok za predaju prvog domaćeg zadatka je **sreda, 14.04.2021.** putem kursa predmeta na Moodle platformi za elektronsko učenje. Tačan termin za predaju će biti naknadno definisan za sve studente. Termin će biti ograničenog vremenskog trajanja.
2. Domaći zadaci će biti testirani i ocenjivani korišćenjem javnih i tajnih testova.
3. Studentima će nekoliko dana pre roka za predaju biti dostupno okruženje za testiranje rešenja domaćeg zadatka na Moodle platformi za elektronsko učenje korišćenjem javnih testova.
4. Prilikom predaje domaćeg zadatka studenti će rešavati i kratak test znanja u vezi rešenja domaćeg zadatka i relevantnog gradiva iz programskog jezika C koje obuhvata temu domaćeg zadatka.
5. Domaći zadaci se rešavaju **samostalno**. Predmetni nastavnici zadržavaju pravo da nakon predaje domaćih zadataka izvrše proveru sličnosti i preuzmu odgovarajuće disciplinske mere.
6. Svi drugi detalji oko predaje, ocenjivanja i odbrane domaćeg zadatka će biti blagovremeno objavljeni.
7. Formula za redni broj problema i koji treba rešavati je sledeća (R – redni broj indeksa, G – poslednje dve cifre godine upisa):  **$i = (R + G) \bmod 8$**
8. Kao rešenje domaćeg zadatka potrebno je predati sadržaj sledeće datoteke:
  - **dz1.c**, koja sadrži izvorni tekst osnovnog programa na programskom jeziku C;

06.04.2021. godine

sa predmeta

0. Napisati program koji iz niza celih brojeva, izbacuje sve elemente koji su deljivi sa  $k$ , a zatim u rezultujućem nizu svaki element zameni zbirom njegovih cifara.

Program treba da:

- 1) Učita dužinu niza, a zatim u novom redu i niz celih brojeva tipa `int`.
- 2) Učita ceo broj  $k$ .
- 3) Ispiše učitani niz. Svaki element treba da bude razdvojen tačno jednim blanko znakom. Nakon poslednjeg elementa ne treba da se ispiše blanko znak.
- 4) Iz niza izbacuje sve elemente koji su deljivi sa  $k$  i ispiše novi niz.
- 5) Svaki element novog niza zameni njegovim zbirom cifara i ispiše rezultujući niz.

### Primeri

<b>Ulaz:</b>	<b>Izlaz:</b>
5 1 2 3 4 5 3	1 2 3 4 5 1 2 4 5 1 2 4 5
7 11 55 24 13 5 0 32 11	11 55 24 13 5 0 32 24 13 5 32 6 4 5 5
6 920 944 546 114 122 8 7	920 944 546 114 122 8 920 944 114 122 8 11 17 6 5 8

1. Napisati program koji iz niza celih brojeva, izbacuje sve susedne elemente koji imaju istu vrednost. Ukoliko dva susedna elementa imaju istu vrednost izbaciti oba elementa. Smatrati da se neće desiti da više od dva uzastopna elementa imaju istu vrednost. Nakon toga u rezultujućem nizu svaki element zameniti najvećim stepenom broja dva koji je manji od njega ili jednak njemu.

Program treba da:

- 1) Učita dužinu niza, a zatim u novom redu i niz celih brojeva tipa `int`.
- 2) Ispiše učitani niz. Svaki element treba da bude razdvojen tačno jednim blanko znakom. Nakon poslednjeg elementa ne treba da se ispiše blanko znak.
- 3) Iz niza izbaciti sve susedne elemente koji su imaju istu vrednost i ispiše novi niz.
- 4) Svaki element novog niza zameni na opisan način i ispiše rezultujući niz.

### Primeri

Ulaz:	Izlaz:
7 1 1 2 2 1 3 5	1 1 2 2 1 3 5 1 3 5 1 2 4
6 15 16 17 17 18 19	15 16 17 17 18 19 15 16 18 19 8 16 16 16
8 2 2 1 3 3 1 2 2	2 2 1 3 3 1 2 2 1 1 1 1

2. Napisati program koji iz niza celih brojeva, izbacuje sve elemente koji predstavljaju stepene broja dva, a zatim u rezultujućem nizu pronade element koji je po apsolutnoj vrednosti najbliži nekom stepenu dvojke. U slučaju da više elemenata zadovoljava zadati uslov, ispisati ih sve u poretku u kojem su zadati u rezultujućem nizu.

Program treba da:

- 1) Učita dužinu niza, a zatim u novom redu i niz celih brojeva tipa `int`.
- 2) Ispiše učitani niz. Svaki element treba da bude razdvojen tačno jednim blanko znakom. Nakon poslednjeg elementa ne treba da se ispiše blanko znak.
- 3) Iz niza izbaci sve elemente koji su stepeni broja dva i ispiše novi niz.
- 4) Pronade i ispiše element rezultujućeg niz koji zadovoljava zadati uslov.

### Primeri

<b>Ulaz:</b>	<b>Izlaz:</b>
5 14 2 8 11 16	14 2 8 11 16 14 11 14
9 7 19 18 6 5 19 3 17 11	7 19 18 6 5 19 3 17 11 7 19 18 6 5 19 3 17 11 7 5 3 17
6 20 13 12 4 16 11	20 13 12 4 16 11 20 13 12 11 13 11

3. Napisati program koji vrši obradu nad dva niza celih brojeva **a** i **b**. Program najpre izbacuje elemente niza **a** sa pozicije **i** ukoliko važi da je  $b[i] < b[i+1]$ . Nakon toga za svaki element niza **a** ispisati 1 ukoliko se element  $a[i]$  predstavlja kao bitski komplement nekog od elemenata iz niza **b**. U suprotnom se ispisuje 0.

Program treba da:

- 1) Učita dužinu nizova **a** i **b**, zatim u novom redu niz **a** celih brojeva tipa **short**, a zatim u novom redu niz **b** celih brojeva tipa **short**
- 2) Ispíše učitani niz **a**. Svaki element treba da bude razdvojen tačno jednim blanko znakom. Nakon poslednjeg elementa ne treba da se ispiše blanko znak.
- 3) Iz niza **a** izbaciti elemente po gore opisanom pravilu i ispiše novi niz.
- 4) Za svaki element novog niza **a** ispiše vrednost koja se dobija po gore opisanom pravilu.

### Primeri

Ulaz:	Izlaz:
6 14 11 -10 11 4 -9 8 -3 7 15 7 1	14 11 -10 11 4 -9 14 11 4 -9 0 0 0 1
9 -3 -8 -2 8 7 7 1 -5 2 8 7 1 2 0 8 3 8 -6	-3 -8 -2 8 7 7 1 -5 2 -3 -8 8 7 -5 2 1 1 0 0 0 0
4 -4 -3 2 -1 0 2 -5 -3	-4 -3 2 -1 -3 -1 1 1

4. Napisati program koji pronalazi i uklanja sve kružnice u ravni koje se međusobno seku. Kružnice se zadaju pomoću tri niza realnih brojeva  $x[i]$ ,  $y[i]$  i  $r[i]$  koji predstavljaju  $x$ ,  $y$  i  $r$  koordinate centra i poluprečnik kružnice. Korisnik zadaje kružnice unošenjem parova  $(x, y, r)$ . Nakon toga program pronalazi i izbacuje iz niza kružnice koje se seku i to tako da kada se naiđe na par kružnica koje se seku izbacuje se ona koja je prva uneta. Smatrati da se kružnice seku ukoliko je suma poluprečnika kružnica veća od ili jednaka rastojanju centara kružnica. Nakon toga je potrebno ispisati kružnicu sa najmanjom udaljenošću od koordinatnog početka. U slučaju više kružnica koje zadovoljavaju zadati uslov, ispisati ih sve u poretku u kojem su zadate u polaznim nizovima.

Program treba da:

- 1) Učita broj kružnica  $n$ .
- 2) Učita  $n$  kružnica sa standardnog ulaza unošenjem parova  $(x, y, r)$ .
- 3) Ispiše učitane kružnice ispisivanjem parova  $(x, y, r)$ . Svaki par ispisati u zasebnom redu. Nakon poslednjeg para ispisati znak za novi red.
- 4) Iz nizova izbace sve kružnice koje se seku i ispišu preostale kružnice na prethodno opisani način.
- 5) Ispiše kružnicu sa najmanjom udaljenošću od koordinatnog početka u obliku uređenog para  $(x, y, r)$ .

### Primeri

Ulaz:	Izlaz:
2 (1, 1, 1) (2, 2, 2)	(1.00, 1.00, 1.00) (2.00, 2.00, 2.00) (2.00, 2.00, 2.00) (2.00, 2.00, 2.00)
7 (-2.14, -4.30, 2.92) (-7.70, 1.68, 1.74) (-9.52, 4.93, 2.72) (7.06, 1.29, 1.16) (2.59, 9.86, 2.20) (-8.30, 7.24, 2.22) (7.88, -5.00, 2.58) (7.88, -5.00, 2.58)	(-2.14, -4.30, 2.92) (-7.70, 1.68, 1.74) (-9.52, 4.93, 2.72) (7.06, 1.29, 1.16) (2.59, 9.86, 2.20) (-8.30, 7.24, 2.22) (7.88, -5.00, 2.58) (-2.14, -4.30, 2.92) (7.06, 1.29, 1.16) (2.59, 9.86, 2.20) (-8.30, 7.24, 2.22) (7.88, -5.00, 2.58) (-2.14, -4.30, 2.92)
5 (3.55, 7.18, 1.83) (7.12, -2.35, 3.40) (-5.27, 1.50, 1.63) (3.59, 6.19, 3.38) (5.56, -0.94, 3.61)	(3.55, 7.18, 1.83) (7.12, -2.35, 3.40) (-5.27, 1.50, 1.63) (3.59, 6.19, 3.38) (5.56, -0.94, 3.61) (-5.27, 1.50, 1.63) (3.59, 6.19, 3.38) (5.56, -0.94, 3.61) (-5.27, 1.50, 1.63)

5. Napisati program koji pronalazi i ispisuje pozicije svih onih elemente u nizu celih brojeva čija se binarna reprezentacija sastoji od naizmeničnih nula ili jedinica, a zatim formira novi niz koji ne sadrži navedene elemente. Elementi novog niza treba da budu u obrnutom poretku u odnosu na polazni niz. Nakon formiranja ispisati rezultujući niz. Opisanu proveru jednog elementa niza vršiti na kompletnoj širini celobrojnog tipa koji se koristi. Iz opisane provere izuzeti vodeće nule, odnosno uzeti u obzir samo širinu koja je neophodna za smeštanje datog broja.

Program treba da:

- 1) Učita dužinu niza, a zatim u novom redu i niz celih brojeva tipa `int`.
- 2) Ispiše učitani niz. Svaki element treba da bude razdvojen tačno jednim blanko znakom. Nakon poslednjeg elementa ne treba da se ispiše blanko znak.
- 3) Ispiše pozicije svih onih elemente koji zadovoljavaju navedeni uslov. Svaki indeks treba da bude razdvojen tačno jednim blanko znakom. Nakon poslednjeg elementa ne treba da se ispiše blanko znak.
- 4) Formira novi niz koji će u obrnutom poretku sadržati elemente polaznog niza koji ne zadovoljavaju navedeni uslov.
- 5) Ispiše novoformirani, rezultujući niz. Svaki element treba da bude razdvojen tačno jednim blanko znakom. Nakon poslednjeg elementa ne treba da se ispiše blanko znak.

### Primeri

<b>Ulaz 0:</b>
11 63 16 433 125 618 534 821 85 51 223 519
<b>Izlaz 0:</b>
63 16 433 125 618 534 821 85 51 223 519 7 519 223 51 821 534 618 125 433 16 63
<b>Ulaz 1:</b>
6 822 611 53 137 113 258
<b>Izlaz 1:</b>
822 611 53 137 113 258  258 113 137 53 611 822
<b>Ulaz 2:</b>
5 1 2 3 4 5
<b>Izlaz 2:</b>
1 2 3 4 5 0 1 4 4 3

6. Napisati program koji formira novi niz na osnovu zadanog niza pozitivnih celih brojeva čiji elementi sadrže prvu sledeću manju vrednost u odnosu na odgovarajući element zadanog niza. Ukoliko takva vrednost ne postoji, upisati nulu. Nakon unošenja niza, proveriti da li su u niz uneti samo pozitivni celi brojevi. Ukoliko nisu, prekinuti izvršavanje i ispisati poruku **NISU POZITIVNI**. U suprotnom ispisati **JESU POZITIVNI** i nastaviti dalje izvršavanje.

Program treba da:

- 1) Učita dužinu niza, a zatim u novom redu i niz celih brojeva tipa `int`.
- 2) Izvrši proveru da li niz sadrži samo pozitivne cele brojeve. Ukoliko ne sadrži, ispisati navedenu poruku i prekinuti izvršavanje programa. Ukoliko sadrži, ispisati navedenu poruku i nastaviti dalje izvršavanje.
- 3) Ispiše učitani niz. Svaki element treba da bude razdvojen tačno jednim blanko znakom. Nakon poslednjeg elementa ne treba da se ispiše blanko znak.
- 4) Formira novi niz na prethodno opisani način.
- 5) Ispiše novoformirani, rezultujući niz.

### Primeri

<b>Ulaz 0:</b>
30 3 1 2 6 1 8 8 5 1 3 9 4 1 9 6 2 3 2 2 2 9 1 2 3 8 7 8 3 3 4
<b>Izlaz 0:</b>
JESU POZITIVNI 3 1 2 6 1 8 8 5 1 3 9 4 1 9 6 2 3 2 2 2 9 1 2 3 8 7 8 3 3 4 1 0 1 1 0 5 5 1 0 1 4 1 0 6 2 1 2 1 1 1 1 0 0 0 7 3 3 0 0 0
<b>Ulaz 1:</b>
20 61 34 57 10 57 50 59 17 31 36 66 84 45 55 35 55 99 23 10 31
<b>Izlaz 1:</b>
JESU POZITIVNI 61 34 57 10 57 50 59 17 31 36 66 84 45 55 35 55 99 23 10 31 34 10 10 0 50 17 17 10 23 35 45 45 35 35 23 23 23 10 0 0
<b>Ulaz 2:</b>
20 2 -1 8 -1 9 7 3 -2 3 4 3 -2 6 2 8 5 0 1 3 7
<b>Izlaz 2:</b>
NISU POZITIVNI

7. Napisati program koji vrši spajanje dva neopadajuće ili nerastuće uređena niza u treći na isti način uređeni niz. Detekciju vrste uređenosti treba obaviti programski. Ukoliko nizovi nisu uređeni na isti način, izvršiti obrtanje elemenata drugog niza, pa nastaviti sa obradom. Nakon unošenja nizova, proveriti za svaki niz da li je uređen. Ukoliko nije, prekinuti izvršavanje i ispisati poruku **NIJE UREĐEN**. U suprotnom ispisati **JESTE UREĐEN** i nastaviti dalje izvršavanje.

Program treba da:

- 1) Učita dužinu prvog niza, a zatim u novom redu i niz celih brojeva tipa `int`.
- 2) Izvrši proveru da li je niz uređen. Ukoliko nije, prekinuti izvršavanje i ispisati poruku **NIJE UREĐEN**. U suprotnom ispisati **JESTE UREĐEN** i nastaviti dalje izvršavanje.
- 3) Učita dužinu drugog niza, a zatim u novom redu i niz celih brojeva tipa `int`.
- 4) Izvrši proveru da li je niz uređen. Ukoliko nije, prekinuti izvršavanje i ispisati poruku **NIJE UREĐEN**. U suprotnom ispisati **JESTE UREĐEN** i nastaviti dalje izvršavanje.
- 5) Ispiše učitanе nizove, svaki u zasebnom redu. Svaki element treba da bude razdvojen tačno jednim blanko znakom. Nakon poslednjeg elementa ne treba da se ispiše blanko znak.
- 6) Ukoliko je potrebno, izvrši obrtanje drugog niza.
- 7) Formira novi niz na prethodno opisani način.
- 8) Ispiše novoformirani, rezultujući niz.

### Primeri

<b>Ulaz 0:</b>
10 9 8 8 6 5 5 4 3 3 2 10 1 1 2 3 6 7 7 8 9 9
<b>Izlaz 0:</b>
JESTE UREDJEN JESTE UREDJEN 9 8 8 6 5 5 4 3 3 2 1 1 2 3 6 7 7 8 9 9 9 9 9 8 8 8 7 7 6 6 5 5 4 3 3 3 2 2 1 1
<b>Ulaz 1:</b>
10 2 3 3 4 5 5 6 8 8 9 10 9 9 8 7 7 6 3 2 1 1
<b>Izlaz 1:</b>
OUTPUT : JESTE UREDJEN JESTE UREDJEN 2 3 3 4 5 5 6 8 8 9 9 9 8 7 7 6 3 2 1 1 1 1 2 2 3 3 3 4 5 5 6 6 7 7 8 8 8 9 9 9
<b>Ulaz 2:</b>
10 1 6 7 8 1 9 2 7 3 9 10 2 3 3 4 5 5 6 8 8 9
<b>Izlaz 2:</b>
NIJE UREDJEN