

## PROGRAMIRANJE 1



**Milena Vujošević Janičić, Jovana Kovačević,  
Danijela Simić, Anđelka Zečević**

# **PROGRAMIRANJE 1**

## **Zbirka zadataka**

**Beograd  
2017.**

Autori:

*dr Milena Vujošević Jančić*, docent na Matematičkom fakultetu u Beogradu

*dr Jovana Kovačević*, docent na Matematičkom fakultetu u Beogradu

*Danijela Simić*, asistent na Matematičkom fakultetu u Beogradu

*Anđelka Zečević*, asistent na Matematičkom fakultetu u Beogradu

PROGRAMIRANJE 1

Zbirka zadataka

# Sadržaj

<b>1</b>	<b>Predstavljanje podataka</b>	<b>vii</b>
1.1	Nizovi . . . . .	vii
1.2	Rešenja . . . . .	xxiv



# 1

## Predstavljanje podataka

### 1.1 Nizovi

**Zadatak 1.1.1** Ako su  $a = (a_1, \dots, a_n)$  i  $b = (b_1, \dots, b_n)$  vektori dimenzije  $n$ , njihov skalarni proizvod je  $a \cdot b = a_1 \cdot b_1 + \dots + a_n \cdot b_n$ . Napisati program koji računa skalarni proizvod dva vektora. Vektori se zadaju kao celobrojni nizovi sa najviše 100 elemenata. Program učitava dimenziju i elemente nizova, a na izlaz ispisuje vrednost skalarnog proizvoda.

#### *Primer 1*

```
INTERAKCIJA SA PROGRAMOM:
Unesite dimenziju vektora:
5
Unesite koordinate vektora a:
8 -2 0 2 4
Unesite koordinate vektora b:
35 12 5 -6 -1
Skalarni proizvod vektora a i b:
240
```

#### *Primer 2*

```
INTERAKCIJA SA PROGRAMOM:
Unesite dimenziju vektora:
3
Unesite koordinate vektora a:
-1 0 1
Unesite koordinate vektora b:
5 5 5
Skalarni proizvod vektora a i b:
0
```

#### *Primer 3*

```
INTERAKCIJA SA PROGRAMOM:
Unesite dimenziju vektora:
120
Greska: Nedozvoljena vrednost!
```

[Rešenje 1.1.1]

**Zadatak 1.1.2** Napisati program koji za učitani niz ispisuje:

(a) elemente niza koji se nalaze na parnim pozicijama.

(b) parne elemente niza.

Pretpostaviti da je dimenzija niza broj koji nije veći od 100.

*Primer 1*

```
INTERAKCIJA SA PROGRAMOM:
Unesite dimenziju niza:
6
Unesite elemente niza:
1 8 2 -5 -13 75
Elementi niza na parnim pozicijama:
1 2 -13
Parni elementi niza:
8 2
```

*Primer 2*

```
INTERAKCIJA SA PROGRAMOM:
Unesite dimenziju niza:
3
Unesite elemente niza:
11 81 -63
Elementi niza na parnim pozicijama:
11 -63
Parni elementi niza:
```

*Primer 3*

```
INTERAKCIJA SA PROGRAMOM:
Unesite dimenziju niza:
-4
Greska: Nedozvoljena vrednost!
```

[Rešenje 1.1.2]

**Zadatak 1.1.3** Napisati program koji za učitani ceo broj, ispisuje broj pojavljivanja svake od cifara u zapisu tog broja. UPUTSTVO: *Za evidenciju broja pojavljivanja svake cifre pojedinačno, koristiti niz.*

*Primer 1*

```
INTERAKCIJA SA PROGRAMOM:
Unesite ceo broj:
2355623
U zapisu broja 2355623, cifra 2 se pojaviljuje 2 puta
U zapisu broja 2355623, cifra 3 se pojaviljuje 2 puta
U zapisu broja 2355623, cifra 5 se pojaviljuje 2 puta
U zapisu broja 2355623, cifra 6 se pojaviljuje 1 puta
```

*Primer 2*

```
INTERAKCIJA SA PROGRAMOM:
Unesite ceo broj:
-39902
U zapisu broja -39902, cifra 0 se pojaviljuje 1 puta
U zapisu broja -39902, cifra 2 se pojaviljuje 1 puta
U zapisu broja -39902, cifra 3 se pojaviljuje 1 puta
U zapisu broja -39902, cifra 9 se pojaviljuje 2 puta
```

[Rešenje 1.1.3]



**Zadatak 1.1.4** Napisati program koji učitava karaktere sa standardnog ulaza sve do kraja ulaza i izračunava koliko se puta u unetom tekstu pojavila svaka od cifara, svako malo slovo i svako veliko slovo. Ispisati broj pojavljivanja samo za karaktere koji su se u unetom tekstu pojavili barem jednom. UPUTSTVO: Za evidenciju broja pojavljivanja cifara, malih i velikih slova koristiti pojedinačne nizove.

#### Primer 1

```
INTERAKCIJA SA PROGRAMOM:
123 abcabcabc 123
Karakter 1 se pojavljuje 2 puta
Karakter 2 se pojavljuje 2 puta
Karakter 3 se pojavljuje 2 puta
Karakter a se pojavljuje 3 puta
Karakter b se pojavljuje 3 puta
Karakter c se pojavljuje 3 puta
```

#### Primer 2

```
INTERAKCIJA SA PROGRAMOM:
Programiranje 1 je zanimljivo!!
Karakter 1 se pojavljuje 1 puta
Karakter a se pojavljuje 3 puta
Karakter e se pojavljuje 2 puta
Karakter g se pojavljuje 1 puta
Karakter i se pojavljuje 3 puta
Karakter j se pojavljuje 3 puta
Karakter l se pojavljuje 1 puta
Karakter m se pojavljuje 2 puta
Karakter n se pojavljuje 2 puta
Karakter o se pojavljuje 2 puta
Karakter r se pojavljuje 3 puta
Karakter v se pojavljuje 1 puta
Karakter z se pojavljuje 1 puta
Karakter P se pojavljuje 1 puta
```

[Rešenje 1.1.4]

**Zadatak 1.1.5** Sa standardnog ulaza se unosi jedna linija teksta. Napisati program koji izračunava i ispisuje koliko puta se pojavilo svako od slova engleskog alfabeta u unetom tekstu. Ne praviti razliku između malih i velikih slova.

#### Primer 1

```
INTERAKCIJA SA PROGRAMOM:
Tasi, tasi, TaNaNa i SVILENA marama....
a:9 b:0 c:0 d:0 e:1 f:0 g:0 h:0 i:4 j:0 k:0 l:1 m:2
n:3 o:0 p:0 q:0 r:1 s:3 t:3 u:0 v:1 w:0 x:0 y:0 z:0
```

#### Primer 2

```
INTERAKCIJA SA PROGRAMOM:
Mihailo Petrovic Alas (6 Maj 1868 - 8 Jun 1943)
a:4 b:0 c:1 d:0 e:1 f:0 g:0 h:1 i:3 j:2 k:0 l:2 m:2
n:1 o:2 p:1 q:0 r:1 s:1 t:1 u:1 v:1 w:0 x:0 y:0 z:0
```

#### Primer 3

```
INTERAKCIJA SA PROGRAMOM:
Alan Matison Tjuring (London, 23. jun 1912 - Cesir, 7. jun 1954)
a:3 b:0 c:1 d:1 e:1 f:0 g:1 h:0 i:3 j:3 k:0 l:2 m:1
n:7 o:3 p:0 q:0 r:2 s:2 t:2 u:3 v:0 w:0 x:0 y:0 z:0
```

[Rešenje 1.1.5]

**Zadatak 1.1.6** Napisati program koji za dva učitana niza  $a$  i  $b$  dimenzije  $n$  formira i na izlaz ispisuje niz  $c$  koji se dobija naizmeničnim raspoređivanjem elemenata nizova  $a$  i  $b$ , tj.  $c = [a_0, b_0, a_1, b_1, \dots, a_{n-1}, b_{n-1}]$ . Pretpostaviti da dimenzija učitanih nizova nije veća od 100.

*Primer 1*

```
|| INTERAKCIJA SA PROGRAMOM:  
|| Unesite dimenziju nizova:  
|| 5  
|| Unesite elemente niza a:  
|| 2 -5 11 4 8  
|| Unesite elemente niza b:  
|| 3 3 9 -1 17  
|| Rezultujući niz:  
|| 2 3 -5 3 11 9 4 -1 8 17
```

*Primer 2*

```
|| INTERAKCIJA SA PROGRAMOM:  
|| Unesite dimenziju nizova:  
|| 105  
|| Greska: Nedozvoljena vrednost!
```

[Rešenje 1.1.6]

**Zadatak 1.1.7** Napisati program koji za dva učitana niza  $a$  i  $b$  dimenzije  $n$  formira i na izlaz ispisuje niz  $c$  čija prva polovina odgovara elementima niza  $b$ , a druga polovina elementima niza  $a$ , tj.  $c = [b_0, b_1, \dots, b_{n-1}, a_0, a_1, \dots, a_{n-1}]$ . Pretpostaviti da dimenzija učitanih nizova nije veća od 100.

*Primer 1*

```
|| INTERAKCIJA SA PROGRAMOM:  
|| Unesite broj n: 3  
|| Unesite elemente niza a: 4 -8 32  
|| Unesite elemente niza b: 5 2 11  
|| 5 2 11 4 -8 32
```

*Primer 2*

```
|| INTERAKCIJA SA PROGRAMOM:  
|| Unesite broj n: 4  
|| Unesite elemente niza a: 1 0 -1 0  
|| Unesite elemente niza b: 5 5 5 3  
|| 5 5 5 3 1 0 -1 0
```

*Primer 3*

```
|| INTERAKCIJA SA PROGRAMOM:  
|| Unesite dimenziju niza: 145  
|| Greska: Nedozvoljena vrednost!
```

[Rešenje 1.1.7]

\* **Zadatak 1.1.8** Sa standardnog ulaza se učitava ceo broj  $n$  manji od 100 i elementi dvaju celobrojnih, sortiranih neopadajuće nizova  $a$  i  $b$  dimenzije  $n$ . Napisati program koji formira i ispisuje niz  $c$  koji se dobija spajanjem nizova  $a$  i  $b$  u treći, takođe sortiran neopadajuće, niz.

### Primer 1

```
INTERAKCIJA SA PROGRAMOM:
Uneti broj elemenata niza: 5
Uneti elemente sortiranog niza:
2 11 28 40 63
Uneti elemente sortiranog niza:
-19 -5 5 11 52
Niz c:
-19 -5 2 5 11 11 28 40 52 63
```

### Primer 2

```
INTERAKCIJA SA PROGRAMOM:
Uneti broj elemenata niza: 3
Uneti elemente sortiranog niza:
-2 4 8
Uneti elemente sortiranog niza:
6 15 19
Niz c:
-2 4 6 8 15 19
```

### Primer 3

```
INTERAKCIJA SA PROGRAMOM:
Uneti broj elemenata niza: 145
Greska: Nedozvoljena vrednost!
```

[Rešenje 1.1.8]

**Zadatak 1.1.9** Napisati program koji sa standardnog ulaza učitava 10 celih brojeva i razdvaja ih na parne i neparne tako što parne brojeve upisuje na početak niza, a neparne brojeve na kraj niza. Ispisati niz dobijen na ovaj način. NAPOMENA: *Nije dozvoljeno koristiti pomoćne nizove.*

### Primer 1

```
INTERAKCIJA SA PROGRAMOM:
Unesite 10 brojeva:
-2 8 11 53 59 20 17 -8 3 14
Rezultujući niz:
14 142 -6 -278 28 34 33 -69 -9 9
```

### Primer 2

```
INTERAKCIJA SA PROGRAMOM:
Unesite 10 brojeva:
9 142 -9 -278 -69 33 34 28 -6 14
Rezultujući niz:
-2 8 14 -8 20 59 17 53 3 11
```

[Rešenje 1.1.9]

**Zadatak 1.1.10** Napisati program koji učitava dimenziju  $n$  celobrojnog niza  $a$  i njegove elemente, i iz niza  $a$  izbacuje sve elemente koji nisu deljivi svojom poslednjom cifrom. Izuzetak su elementi čija je poslednja cifra 0 koje treba zadržati. Program treba da ispiše izmenjeni niz na standardni izlaz. Niz  $a$  sadrži najviše 100 elemenata.

### Primer 1

```
INTERAKCIJA SA PROGRAMOM:
Unesite dimenziju niza:
9
Unesite elemente niza a:
173 -25 23 7 17 25 34 61 -4612
Niz a nakon izmene:
-25 7 25 61 -4612
```

### Primer 2

```
INTERAKCIJA SA PROGRAMOM:
Unesite dimenziju niza:
0
Greska: Nedozvoljena vrednost!
```

**Zadatak 1.1.11** Napisati program koji u nizu dužine  $n$  (broj manji od 100) čiji se elementi učitavaju sa ulaza eliminiše sve brojeve koji nisu deljivi svojim indeksom. Niz reorganizovati tako da nema *rupa* koje su nastale eliminacijom elemenata i ispisati na standardni izlaz. NAPOMENA: *Nulti element niza treba zadržati jer nije dozvoljeno deljenje nulom.*

*Primer 1*

```

|| INTERAKCIJA SA PROGRAMOM:
|| Unesite broj elemenata niza: 10
|| Unesite elemente niza:
|| 4 2 1 6 7 8 10 2 16 3
|| 4 2 6 16

```

**Zadatak 1.1.12** Takmičari na Beogradskom maratonu su označeni rednim brojevima počevši od 0, a vreme za koje su istrčali maraton je dato minutima. Ovi podaci zadati su nizom celih brojeva, pri čemu indeks niza označava redni broj takmičara, a vrednosti u nizu označavaju vreme trčanja. Napisati funkcije za rad sa ovim nizom.

- (a) Napisati funkciju `void ucitaj(int a[], int n)` koja učitava elemente niza  $a$  dimenzije  $n$ .
- (b) Napisati funkciju `void stampaj(int a[], int n)` koja štampa elemente niza  $a$  dimenzije  $n$ .
- (c) Napisati funkciju `int suma(int a[], int n)` koja računa i vraća ukupno vreme trčanja svih takmičara.
- (d) Napisati funkciju `float prosek(int a[], int n)` koja računa i vraća prosečno vreme (aritmetičku sredinu) trčanja takmičara.
- (e) Napisati funkciju `int maksimum(int a[], int n)` koja izračunava i vraća najduže vreme trčanja takmičara.
- (f) Napisati funkciju `int pozicija_minimum(int a[], int n)` koja vraća redni broj pobednika Beogradskog maratona, tj. onog takmičara koji je najkraće trčao. U slučaju da ima više takvih takmičara, vratiti onog sa najmanjim indeksom.

Napisati program koji testira rad zadatih funkcija. Maksimalan broj takmičara je 1000.

### Primer 1

```
INTERAKCIJA SA PROGRAMOM:
Unesite dimenziju niza:
5
19 47 27 34 16
Vreme trcanja takmicara: 19 47 27 34 16
Ukupno vreme: 143
Prosecno vreme trcanja: 28.6
Maksimalno vreme trcanja: 47
Indeks pobednika: 4
```

### Primer 2

```
INTERAKCIJA SA PROGRAMOM:
Unesite dimenziju niza:
-5
Greska: Nedozvoljena vrednost!
```

[Rešenje [1.1.12](#)]

**Zadatak 1.1.13** Elementi niza celih brojeva su podaci o nadmorskim visinama u nekom području sveta. Na kartografskoj mapi su indeksima označene različite tačke, a u nizu je svakom indeksu pridružen neki ceo broj (odnosno, nadmorska visina). Napisati funkcije za rad sa ovim nizom.

- (a) Napisati funkciju koja proverava da li niz sadrži zadatu nadmorsku visinu  $m$ . Povratna vrednost funkcije je 1 ako je vrednost sadržana u nizu ili 0 ako nije.
- (b) Napisati funkciju koja vraća vrednost prve pozicije na kojoj se nalazi element koji ima nadmorsku visinu  $m$  ili  $-1$  ukoliko element nije u nizu.
- (c) Napisati funkciju koja vraća vrednost poslednje pozicije na kojoj se nalazi element koji ima nadmorsku visinu  $m$  ili  $-1$  ukoliko element nije u nizu.

Napisati i program koji testira rad napisanih funkcija za uneti broj  $m$  i niz čija dimenzija nije veća od 100.

### Primer 1

```
INTERAKCIJA SA PROGRAMOM:
Unesite dimenziju niza:
7
800 1100 -200 1400 -200 1100 800
Ucitani niz: 800 1100 -200 1400 -200 1100 800
Unesite jedan ceo broj:
1100
Niz sadrzi element cija je vrednost 1100.
Indeks njegovog prvog pojavljivanja u nizu je 1.
Indeks njegovog poslednjeg pojavljivanja u nizu je 5.
```

### Primer 2

```
INTERAKCIJA SA PROGRAMOM:
Unesite dimenziju niza:
-5
Greska: Nedozvoljena vrednost!
```

[Rešenje ??]

**Zadatak 1.1.14** Napisati funkcije za rad sa nizovima celih brojeva.

- (a) Napisati funkciju koja obrće elemente niza.
- (b) Napisati funkciju koja rotira niz ciklično za jedno mesto u levo.
- (c) Napisati funkciju koja rotira niz ciklično za  $k$  mesta u levo.

Napisati i program koji testira rad napisanih funkcija za uneti broj  $m$  i niz čija dimenzija nije veća od 100.

*Primer 1*

```
INTERAKCIJA SA PROGRAMOM:
Unesite dimenziju niza:
6
7 -3 11 783 26 -19
Elementi niza nakon obrtanja:
-17 28 785 13 -1 9
Elementi niza nakon rotiranja za 1 mesto ulevo:
28 785 13 -1 9 -17
Unesite jedan pozitivan ceo broj:
3
Elementi niza nakon rotiranja za 3 mesto ulevo:
-1 9 -17 28 785 13
```

*Primer 2*

```
INTERAKCIJA SA PROGRAMOM:
Unesite dimenziju niza:
252
Greska: Nedozvoljena vrednost!
```

[Rešenje 1.1.14]

**Zadatak 1.1.15** Napisati program koji transformiše uneti niz tako što kvadrira sve negativne elemente niza. Pretpostaviti da je dimenzija niza broj koji nije veći od 100.

*Primer 1*

```
INTERAKCIJA SA PROGRAMOM:
Unesite broj elemenata niza: 6
Unesite elemente niza:
12.34 -6 1 8 32.4 -16
12.34 36 1 8 32.4 256
```

*Primer 2*

```
INTERAKCIJA SA PROGRAMOM:
Unesite broj elemenata niza: 9
Unesite elemente niza:
-8.25 6 17 2 -1.5 1 -7 2.65 -125.2
68.0625 6 17 2 2.25 1 49 2.65 15675.04
```

*Primer 3*

```
INTERAKCIJA SA PROGRAMOM:
Unesite broj elemenata niza: 4
Unesite elemente niza:
9.53 5 1 4.89
9.53 5 1 4.89
```

[Rešenje 1.1.15]

**Zadatak 1.1.16** Sa standardnog ulaza se učitava dimenzija niza, elementi niza i jedan ceo broj  $k$ . Napisati program koji štampa indekse elemenata koji su deljivi sa  $k$ . Pretpostaviti da je dimenzija niza broj koji nije veći od 100.

*Primer 1*

```
INTERAKCIJA SA PROGRAMOM:
Unesite dimenziju niza: 4
Unesite elemente niza: 10 14 86 20
Unesite broj k: 5
0 3
```

*Primer 2*

```
INTERAKCIJA SA PROGRAMOM:
Unesite dimenziju niza: 4
Unesite elemente niza: 6 14 8 9
Unesite broj k: 5
U nizu nema elemenata koji su deljivi brojem 5!
```

*Primer 3*

```
INTERAKCIJA SA PROGRAMOM:
Unesite dimenziju niza: 6
Unesite elemente niza: 8 9 11 -4 8 11
Unesite broj k: 2
0 3 4
```

[Rešenje 1.1.16]

**Zadatak 1.1.17** Napisati program koji učitava dimenziju i elemente niza i štampa niz u kojem su najveći i najmanji element niza razmenili mesta. Ukoliko se najmanji ili najveći element više puta pojavljuju u nizu, uzeti u obzir njihova prva pojavljivanja. Pretpostaviti da je dimenzija niza broj koji nije veći od 100.

*Primer 1*

```
INTERAKCIJA SA PROGRAMOM:
Unesite dimenziju niza: 5
Unesite elemente niza: 8 -2 11 19 4
8 19 11 -2 4
```

*Primer 2*

```
INTERAKCIJA SA PROGRAMOM:
Unesite dimenziju niza: 10
Unesite elemente niza:
46 -2 51 8 -5 66 2 8 3 14
46 -2 51 8 66 -5 2 8 3 14
```

*Primer 3*

```
INTERAKCIJA SA PROGRAMOM:
Unesite dimenziju niza: 145
Greska: Nedozvoljena vrednost!
```

[Rešenje 1.1.17]

**Zadatak 1.1.18** Napisati funkciju `int min_max(int a[], int n)` koja pronalazi indekse najmanjeg i najvećeg elementa u nizu  $a$  dimenzije  $n$  koristeći samo jedan prolaz kroz niz. Funkcija kao povratnu vrednost vraća manji od ta dva indeksa. Napisati program koji testira ovu funkciju za učitane nizove celih brojeva maksimalne dužine 100 elemenata.

### Primer 1

```
INTERAKCIJA SA PROGRAMOM:
Unesite broj elemenata niza:
7
Unesite elemente niza:
5 8 -4 11 17 89 1
2
```

### Primer 2

```
INTERAKCIJA SA PROGRAMOM:
Unesite broj elemenata niza:
3
Unesite elemente niza:
9 11 6
1
```

### Primer 3

```
INTERAKCIJA SA PROGRAMOM:
Unesite broj elemenata niza:
-45
Greska: Nedozvoljena vrednost!
```

**Zadatak 1.1.19** Napisati program koji učitane karaktere (najviše njih 100, učitavaju se sve do pojave karaktera \*) ispisuje u redosledu suprotnom od redosleda čitanja.

### Primer 1

```
INTERAKCIJA SA PROGRAMOM:
Unesite karakter: a
Unesite karakter: 8
Unesite karakter: 5
Unesite karakter: Y
Unesite karakter: I
Unesite karakter: o
Unesite karakter: ?
Unesite karakter: *
? o I Y 5 8 a
```

### Primer 2

```
INTERAKCIJA SA PROGRAMOM:
Unesite karakter: g
Unesite karakter: g
Unesite karakter: 2
Unesite karakter: 2
Unesite karakter: )
Unesite karakter: )
Unesite karakter: *
) ) 2 2 g g
```

### Primer 3

```
INTERAKCIJA SA PROGRAMOM:
Unesite karakter: U
Unesite karakter: 4
Unesite karakter: a
Unesite karakter: u
Unesite karakter: *
u a 4 U
```

[Rešenje [1.1.19](#)]

### Zadatak 1.1.20

Sa standardnog ulaza se unosi broj elemenata niza  $a$  i njegovi elementi. Napisati program koji od datog niza formira niz  $b$  u koji ulaze elementi niza  $a$  koji se pojavljuju tačno tri puta. Pretpostaviti da će uneti niz imati najviše 100 elemenata. Pretpostaviti da je dimenzija niza broj koji nije veći od 100.



### Primer 1

```
INTERAKCIJA SA PROGRAMOM:
Unesite broj elemenata niza: 8
Unesite elemente niza a:
4 11 4 6 8 4 6 6
Elementi niza b: 4 6
```

### Primer 2

```
INTERAKCIJA SA PROGRAMOM:
Unesite broj elemenata niza: 13
Unesite elemente niza a:
-8 26 7 2 1 1 7 2 2 2 7 5 1
Elementi niza b: 7 1
```

### Primer 3

```
INTERAKCIJA SA PROGRAMOM:
Unesite broj elemenata niza: 2
Unesite elemente niza a:
9 5
Elementi niza b:
```

[Rešenje 1.1.20]

**Zadatak 1.1.21** Napisati funkciju `int sadrzi_bar_dva(int a[], int na, int b[], int nb)` koja proverava da li niz  $a$  dužine  $na$  sadrži barem dva broja koja se pojavljuju u nizu  $b$  dužine  $nb$ . Napisati i program koji učitava redom dimenzije i elemente nizova  $a$  i  $b$  i ispisuje da li uneti nizovi ispunjavaju traženo svojstvo. Pretpostaviti da će uneti nizovi imati najviše 100 elemenata.

### Primer 1

```
INTERAKCIJA SA PROGRAMOM:
Unesite broj elemenata niza a:
5
Unesite elemente niza a:
5 8 7 -2 6
Unesite broj elemenata niza b:
6
Unesite elemente niza b:
11 -11 7 -7 6
Svojstvo je ispunjeno.
```

**Zadatak 1.1.22** Sa standardnog ulaza se, redom, učitavaju dimenzije i elementi dva niza,  $a$  i  $b$ . Napisati program koji određuje i ispisuje njihovu uniju, presek i razliku (redosled prikaza elemenata nije bitan). Pretpostaviti da će uneti nizovi imati najviše 100 elemenata.

### Primer 1

```
INTERAKCIJA SA PROGRAMOM:
Unesite broj elemenata niza a: 5
Unesite elemente niza a: 2 8 1 5 2
Unesite broj elemenata niza b: 3
Unesite elemente niza b: 5 7 8
Unija: 2 8 1 5 2 5 7 8
Presek: 5
Razlika: 2 1 2
```

### Primer 2

```
INTERAKCIJA SA PROGRAMOM:
Unesite broj elemenata niza a: 3
Unesite elemente niza a: 11 4 4
Unesite broj elemenata niza b: 2
Unesite elemente niza b: 18 9
Unija: 11 4 4 18 9
Presek:
Razlika: 11 4 4
```

### Primer 3

```
INTERAKCIJA SA PROGRAMOM:
Unesite broj elemenata niza a: 6
Unesite elemente niza a: 12 7 9 12 5 1
Unesite broj elemenata niza b: 4
Unesite elemente niza b: 1 12 22 12
Unija: 12 7 9 12 5 1 1 12 22 12
Presek: 12 12 1
Razlika: 7 9 5
```

[Rešenje 1.1.22]

**Zadatak 1.1.23** Napisati program koji za učitani niz formira i ispisuje niz koji se dobija izbacivanjem svih neparnih elemenata niza. Zadatak rešiti na dva načina: korišćenjem pomoćnog niza i transformacijom polaznog niza. Pretpostaviti da će uneti niz imati najviše 100 elemenata.

### Primer 1

```
INTERAKCIJA SA PROGRAMOM:
Unesite broj elemenata niza: 4
Unesite elemente niza: 8 9 15 12
8 12
```

### Primer 2

```
INTERAKCIJA SA PROGRAMOM:
Unesite broj elemenata niza: 6
Unesite elemente niza: 21 5 3 22 19 188
22 188
```

### Primer 3

```
INTERAKCIJA SA PROGRAMOM:
Unesite broj elemenata niza: 4
Unesite elemente niza: 133 129 121 101
```

### Primer 4

```
INTERAKCIJA SA PROGRAMOM:
Unesite broj elemenata niza: 8
Unesite elemente niza: 15 -22 -23 13 18 46 14 -31
-22 18 46 14
```

[Rešenje 1.1.23]

**Zadatak 1.1.24** Napisati program koji za učitani niz ispisuje niz koji se dobija izbacivanjem svih elemenata koji su prosti brojevi. Zadatak rešiti na dva načina: korišćenjem pomoćnog niza i transformacijom polaznog niza. Pretpostaviti da će uneti niz imati najviše 100 elemenata. NAPOMENA: *Broj 1 nije prost.*

*Primer 1*

```
INTERAKCIJA SA PROGRAMOM:
Unesite broj elemenata niza: 5
Unesite elemente niza: 11 5 6 48 8
6 48 8
```

*Primer 2*

```
INTERAKCIJA SA PROGRAMOM:
Unesite broj elemenata niza: 4
Unesite elemente niza: 11 5 19 21
21
```

*Primer 3*

```
INTERAKCIJA SA PROGRAMOM:
Unesite broj elemenata niza: 5
Unesite elemente niza: 12 18 9 31 7
12 18 9
```

*Primer 4*

```
INTERAKCIJA SA PROGRAMOM:
Unesite broj elemenata niza: 3
Unesite elemente niza: -31 11 -19
```

*Primer 5*

```
INTERAKCIJA SA PROGRAMOM:
Unesite broj elemenata niza: 5
Unesite elemente niza: -2 15 -11 8 7
15 8
```

[Rešenje 1.1.24]

**Zadatak 1.1.25** Napisati funkciju `int prebrojavanje(int a[], int n)` koja izračunava broj elemenata celobrojnog niza  $a$  dužine  $n$  koji su manji od poslednjeg elementa niza. Napisati i program koji testira rad funkcije. Pretpostaviti da dužina niza neće biti veća od 100.

*Primer 1*

```
INTERAKCIJA SA PROGRAMOM:
Unesite broj elemenata niza: 4
Unesite elemente niza: 11 2 4 9
2
```

*Primer 2*

```
INTERAKCIJA SA PROGRAMOM:
Unesite broj elemenata niza: 7
Unesite elemente niza: 7 2 1 14 65 2 8
4
```

*Primer 3*

```
INTERAKCIJA SA PROGRAMOM:
Unesite broj elemenata niza: 5
Unesite elemente niza: 25 18 29 30 14
0
```

[Rešenje 1.1.25]

**Zadatak 1.1.26** Napisati funkciju `int prebrojavanje(int a[], int n)` koja izračunava broj parnih elemenata niza celih brojeva  $a$  dužine  $n$  koji pret-hode maksimalnom elementu niza. Napisati i program koji testira rad funkcije. Pretpostaviti da dužina niza neće biti veća od 100.

*Primer 1*

```
|| INTERAKCIJA SA PROGRAMOM:
|| Unesite broj elemenata niza: 4
|| Unesite elemente niza: 11 2 4 9
|| 0
```

*Primer 2*

```
|| INTERAKCIJA SA PROGRAMOM:
|| Unesite broj elemenata niza: 7
|| Unesite elemente niza: 7 2 1 14 65 2 8
|| 2
```

*Primer 3*

```
|| INTERAKCIJA SA PROGRAMOM:
|| Unesite broj elemenata niza: 5
|| Unesite elemente niza: 25 18 29 30 14
|| 1
```

[Rešenje 1.1.26]

**Zadatak 1.1.27** Napisati funkciju `int cifre(char s[], int n)` koja izraču-nava broj cifara u nizu karaktera  $a$  dužine  $n$ . Napisati program koji za karaktere koji se unose u zasebnim redovima ispisuje broj unetih cifara. Pretpostaviti da dužina niza neće biti veća od 100.

*Primer 1*

```
|| INTERAKCIJA SA PROGRAMOM:
|| Unesite broj elemenata niza: 5
|| Unesite elemente niza:
|| 4
|| +
|| A
|| u
|| 8
|| Broj cifara je: 2
```

*Primer 2*

```
|| INTERAKCIJA SA PROGRAMOM:
|| Unesite broj elemenata niza: 7
|| Unesite elemente niza:
|| J
|| M
|| a
|| 5
|| 5
|| -
|| 2
|| Broj cifara je: 3
```

*Primer 3*

```
|| INTERAKCIJA SA PROGRAMOM:
|| Unesite broj elemenata niza: 3
|| Unesite elemente niza:
|| e
|| k
|| F
|| Broj cifara je: 0
```

[Rešenje 1.1.27]

**Zadatak 1.1.28** Napisati funkciju `int zbir(int a[], int n, int i, int j)` koja računa zbir elemenata niza celih brojeva  $a$  dužine  $n$  od pozicije  $i$  do pozicije  $j$ . Napisati i program koji testira rad funkcije. Pretpostaviti da dužina niza neće biti veća od 100.

*Primer 1*

```
INTERAKCIJA SA PROGRAMOM:
Unesite broj elemenata niza: 5
Unesite elemente niza: 11 5 6 48 8
Unesite vrednosti za i i j: 0 2
Zbir je: 22
```

*Primer 2*

```
INTERAKCIJA SA PROGRAMOM:
Unesite broj elemenata niza: 3
Unesite elemente niza: -2 8 1
Unesite vrednosti za i i j: 8 12
Greska: Nekorektne vrednosti granica!
```

*Primer 3*

```
INTERAKCIJA SA PROGRAMOM:
Unesite broj elemenata niza: 7
Unesite elemente niza: -2 5 9 11 6 -3 -4
Unesite vrednosti za i i j: 2 5
Zbir: 23
```

[Rešenje [1.1.28](#)]

**Zadatak 1.1.29** Napisati funkciju `float zbir_pozitivnih(float a[], int n, int k)` koja izračunava zbir prvih  $k$  pozitivnih elemenata realnog niza  $a$  dužine  $n$ . Napisati i program koji testira rad funkcije. Pretpostaviti da dužina niza neće biti veća od 100.

*Primer 1*

```
INTERAKCIJA SA PROGRAMOM:
Unesite broj elemenata niza: 8
Unesite elemente niza:
2.34 1 -12.7 5.2 -8 -6.2 7 14.2
Unesite vrednost za k: 3
Zbir je: 8.54
```

*Primer 2*

```
INTERAKCIJA SA PROGRAMOM:
Unesite broj elemenata niza: 3
Unesite elemente niza:
-6.598 -8.14 -15
Unesite vrednost za k: 4
Zbir je: 0.00
```

*Primer 3*

```
INTERAKCIJA SA PROGRAMOM:
Unesite broj elemenata niza: 7
Unesite elemente niza:
-35.11 5.29 -1.98 12.1 12.2 -3.33 -4.17
Unesite vrednost za k: 15
Zbir: 29.59
```

[Rešenje [1.1.29](#)]

**Zadatak 1.1.30** Napisati funkciju `void kvadriranje(float a[], int n)` koja kvadrira elemente realnog niza  $a$  dužine  $n$  koji se nalaze na parnim pozicijama. Napisati program koji transformiše na ovaj način uneti niz. Pretpostaviti da dužina niza neće biti veća od 100.

*Primer 1*

```
INTERAKCIJA SA PROGRAMOM:
Unesite broj elemenata niza: 8
Unesite elemente niza:
2.34 1 -12.7 5.2 -8 -6.2 7 14.2
5.4756 1 161.29 5.2 64 -6.2 49 14.2
```

*Primer 2*

```
INTERAKCIJA SA PROGRAMOM:
Unesite broj elemenata niza: 3
Unesite elemente niza:
-6 -8.14 -15
36 -8.14 225
```

*Primer 3*

```
INTERAKCIJA SA PROGRAMOM:
Unesite broj elemenata niza: 1
Unesite elemente niza:
-35.11
1232.71
```

[Rešenje 1.1.30]

**Zadatak 1.1.31** Napisati funkciju `int blizu_3(int a[], int n)` koja pronalazi i vraća indeks elementa niza koji je po vrednosti najbliži aritmetičkoj sredini onih elemenata niza koji su deljivi brojem tri. Napisati program koji testira rad funkcije. Pretpostaviti da dužina niza neće biti veća od 100.

*Primer 1*

```
INTERAKCIJA SA PROGRAMOM:
Unesite broj elemenata niza: 5
Unesite elemente niza:
1 2 3 4 5
2
```

*Primer 2*

```
INTERAKCIJA SA PROGRAMOM:
Unesite broj elemenata niza: 5
Unesite elemente niza:
3 6 2 4 7
3
```

**Zadatak 1.1.32** Napisati funkcije za rad sa nizovima celih brojeva.

- Napisati funkciju koja izbacuje poslednji element niza.
- Napisati funkciju koja izbacuje prvi element niza. Zadatak rešiti na dva načina: čuvanjem redosleda elemenata i premeštanjem poslednjeg elementa niza na upražnjenu poziciju.
- Napisati funkciju koja izbacuje element sa date pozicije  $k$ .
- Napisati funkciju koja izbacuje sva pojavljivanja datog elementa  $x$  iz niza.
- Napisati funkciju koja ubacuje dati element  $x$  na kraj niza.

- (f) Napisati funkciju koja ubacuje dati element  $x$  na početak niza.
- (g) Napisati funkciju koja ubacuje dati element  $x$  na datu poziciju  $k$ .

Napisati program koji testira rad zadatih funkcija. Sa standardnog ulaza učitati dimenziju niza (broj ne veći od 100).

#### Primer 1

```
INTERAKCIJA SA PROGRAMOM:
Unesite dimenziju niza:
8
2 5 -2 16 33 19 8 11
Niz posle izbacivanja poslednjeg elementa: 2 5 -2 16 33 5 8
Niz nakon izbacivanja prvog elementa: 5 -2 16 33 5 8
Unesite poziciju elementa za izbacivanje:
3
Niz nakon izbacivanja 3. elementa: 5 -2 16 5 8
Unesite element cije pojavljivanje treba izbaciti:
5
Niz nakon izbacivanja elementa 5: -2 16 8
Unesite element koji treba ubaciti u niz:
19
Niz nakon ubacivanja elementa 19 na kraj: -2 16 8 19
Niz nakon ubacivanja elementa 19 na pocetak: 19 -2 16 8 19
Unesite poziciju na koju treba ubaciti element:
2
Niz nakon ubacivanja elementa 19 na poziciju 2: 19 -2 19 16 8 19
```

**Zadatak 1.1.33** Napisati funkcije za rad sa nizovima celih brojeva.

- (a) Napisati funkciju koja određuje dužinu najduže serije jednakih uzastopnih elemenata u datom nizu brojeva.
- (b) Napisati funkciju koja određuje dužinu najvećeg neopadajućeg podniza datog niza celih brojeva.
- (c) Napisati funkciju koja određuje da li se jedan niz javlja kao podniz uzastopnih elemenata drugog niza.
- (d) Napisati funkciju koja određuje da li se jedan niz javlja kao podniz elemenata drugog niza (elementi ne moraju da budu uzastopni, ali je redosled pojavljivanja isti).
- (e) Napisati funkciju koja izbacuje višestruka pojavljivanja elemenata iz datog niza brojeva. Zadatak rešiti na dva načina: zadržavanjem prvog pojavljivanje elementa i zadržavanjem poslednjeg pojavljivanje elementa.

**Zadatak 1.1.34** Napisati funkciju koja iz zadatog niza izbacuje sve elemente koji su deljivi svojim indeksom. Niz reorganizovati tako da nema *rupa* koje su

nastale izbacivanjem elemenata. Povratna vrednost funkcije je nova dimenzija niza. Napisati program koji za učitni niz (dimenzije manje od 100) ispisuje niz dobijen nakon poziva funkcije. NAPOMENA: *Element na nultoj poziciji niza zadržati jer nije dozvoljeno deljenje nulom.*

*Primer 1*

```
|| INTERAKCIJA SA PROGRAMOM:  
|| Unesite broj elemenata niza: 10  
|| Unesite elemente niza:  
|| 4 2 1 6 7 10 8 2 16 27  
|| 4 1 7 8 2 16
```

**Zadatak 1.1.35** Za celobrojni niz  $a$  dimenzije  $n$  kažemo da je *permutacija* ako sadrži sve brojeve od 1 do  $n$ .

- (a) Napisati funkciju `void brojanje(int a[], int b[], int n)` koja na osnovu celobrojnog niza  $a$  dimenzije  $n$  formira niz  $b$  tako što  $i$ -ti element niza  $b$  odgovara broju pojavljivanja vrednosti  $i$  u nizu  $a$ .
- (b) Napisati funkciju `int permutacija(int a[], int n)` koja proverava da li je zadati niz permutacija. Funkcija vraća vrednost 1 ako je svojstvo ispunjeno, odnosno 0 ako nije. UPUTSTVO: *Koristiti funkciju brojanje iz tačke (a).*

Napisati program koji sa standardnog ulaza učitava dimenziju niza (broj manji od 100) i elemente niza i ispisuje da li je uneti niz permutacija ili ne.

*Primer 1*

```
|| INTERAKCIJA SA PROGRAMOM:  
|| Unesite broj elemenata niza: 5  
|| Unesite elemente niza:  
|| 1 5 4 3 2  
|| Uneti niz je permutacija.
```

*Primer 2*

```
|| INTERAKCIJA SA PROGRAMOM:  
|| Unesite broj elemenata niza: 6  
|| Unesite elemente niza:  
|| 2 3 3 1 1 5  
|| Uneti niz nije permutacija.
```

[Rešenje 1.1.35]

## 1.2 Rešenja

### Rešenje 1.1.1

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



```

4  /* Predprocesorska direktiva kojom se definise maksimalni broj
   *   elemenata niza */
   #define MAX 100

6  int main()
   {
8     int a[MAX];
     int b[MAX];
10    int n;
     int i;
12    int skalarni_proizvod;

14
     /* Ucitava se dimenzija vektora i proverava njena ispravnost */
16    printf("Unesite dimenziju vektora: ");
     scanf("%d", &n);
18    if (n<1 || n>100)
     {
20        printf("Nedozvoljena vrednost!\n");
         return -1;
22    }

24    /* Ucitavaju se koordinate vektora */
     printf("Unesite koordinate vektora a: ");
26    for (i=0; i<n; i++)
     {
28        scanf("%d", &a[i]);
     }

30
     printf("Unesite koordinate vektora b: ");
32    for (i=0; i<n; i++)
     {
34        scanf("%d", &b[i]);
     }

36
     /* Izracunava se skalarni proizvod po zadataj formuli */
38    skalarni_proizvod=0;

40    for (i=0; i<n; i++)
         skalarni_proizvod = skalarni_proizvod + a[i]*b[i];
42
     /* I ispisuje se njegova vrednost */
44    printf("Skalarni proizvod vektora a i b: %d\n",skalarni_proizvod);

46    return 0;
   }

```

## Rešenje 1.1.2

```

#include <stdio.h>

2

```

```

4  #define MAX 100
6  int main()
6  {
8      int a[MAX];
8      int n;
8      int i;
10
12     /* Ucitava se dimenzija niza i proverava se njena ispravnost. */
12     printf("Unesi dimenziju niza:\n");
12     scanf("%d", &n);
14     if (n<1 || n>MAX)
14     {
16         printf("Nedozvoljena vrednost!\n");
16         return -1;
18     }
20
22     /* Ucitavaju se elementi niza */
22     printf("Unesi elemente niza:\n");
22     for (i=0; i<n; i++)
22     {
24         scanf("%d", &a[i]);
24     }
26
28     /* Ispisuju se elementi niza na parnim pozicijama */
28     printf("Elementi niza na parnim pozicijama:\n");
28     for (i=0; i<n; i+=2)
28     {
30         printf("%d ", a[i]);
30     }
32     printf("\n");
34
36     /* Ispisuju se parni elementi niza */
36     printf("Parni elementi niza:\n");
36     for (i=0; i<n; i++){
38         if (a[i]%2==0){
38             printf("%d ", a[i]);
38         }
40     }
42     printf("\n");
44
46     return 0;
46 }

```

### Rešenje 1.1.3

```

1  #include<stdio.h>
1  #include<stdlib.h>
3
3  int main()

```

```

5 {
6     int x;
7     int brojac[10];
8     char cifra;
9     int original;
10    int i;
11
12    /* Ucitava se ceo broj sa standardnog ulaza */
13    printf("Unesite ceo broj:\n");
14    scanf("%d",&x);
15
16    /* Cuva se njegova originalna vrednost zbog finalnog ispisa */
17    original = x;
18
19    /* I nadalje posmatra apsolutna vrednost */
20    x = abs(x);
21
22    /* Svaki element niza brojac predstavlja broj cifara za jednu od
23       cifara:
24       brojac[0] predstavlja broj nula u zapisu broja x
25       brojac[1] predstavlja broj jedinica u zapisu broja x
26       ...
27       brojac[9] predstavlja broj devetki u zapisu broja x
28    */
29
30    /* Brojac se na pocetku inicijalizuje nulama */
31    for(i=0;i<10;i++){
32        brojac[i]=0;
33    }
34
35    /* Sve dok ima cifara u zapisu broja x */
36    do
37    {
38        /* Izdvaja se krajnja desna cifra */
39        cifra = x%10;
40
41        /* Uvecava se njen broj pojavljivanja */
42        brojac[cifra]++;
43
44        /* I prelazi se na analiziranje sledece cifre */
45        x/=10;
46    } while(x);
47
48    /* Ispisuju se informacije o ciframa koje se nalaze u zapisu broja
49       x */
50    for(i=0; i<10; i++){
51        if(brojac[i]){
52            printf("U zapisu broja %d, cifra %d se pojavljuje %d puta\n",
53                original, i, brojac[i]);
54        }
55    }

```

```
    }  
55  
57     return 0;  
}
```

### Rešenje 1.1.4

```
#include <stdio.h>  
2  
/* Funkcija za ispis elemenata niza. */  
4 void ispis(int niz[], int n, char c)  
{  
6     int i;  
  
8     for(i = 0; i < n; i++){  
        if (niz[i]!=0)  
10         printf("Karakter %c se pojavljuje %d puta\n", c + i, niz[i]);  
    }  
12 }  
  
14 int main()  
{  
16     /* Niz u kojem ce se cuvati informacije o broju pojavljivanja  
        cifara */  
    int cifre[10];  
18  
    /* Niz u kojem ce se cuvati informacije o broju pojavljivanja malih  
        slova */  
20    int mala_slova[26];  
  
22    /* Niz u kojem ce se cuvati informacije o broju pojavljivanja  
        velikih slova */  
    int velika_slova[26];  
24  
    int c, i;  
  
26    /* Brojaci karaktera se na pocetku inicijalizuju nulama */  
28    for(i=0;i<10;i++){  
        cifre[i]=0;  
30    }  
  
32    for(i=0;i<26;i++){  
        {  
34        mala_slova[i]=0;  
        velika_slova[i]=0;  
36    }  
    }  
  
38    /* Ucitavaju se karakteri sve do kraja ulaza */  
    while((c = getchar()) != EOF)  
40    {
```

```

42  /* Ako je procitani karakter veliko slovo ... */
    if (c>='A' && c<='Z'){
        velika_slova[c-'A']++;
44  }
    else{
46      /* Ako je procitani karakter malo slovo ... */
        if (c>='a' && c<='z'){
48          mala_slova[c-'a']++;
        }
        else{
50          /* Ako je procitani karakter cifra ... */
52          if(c >='0' && c <= '9'){
              cifre[c-'0']++;
54          }
        }
56    }
}

58
/* Ispisuju se trazene informacije */
60 ispis(cifre, 10, '0');
    ispis(mala_slova, 26, 'a');
62 ispis(velika_slova, 26, 'A');

64 return 0;
}

```

## Rešenje 1.1.5

```

#include <stdio.h>
2  #include <ctype.h>

4  /* Funkcija za ispis elemenata niza. */
void ispis(int niz[], int n)
6  {
    int i;

8    for(i = 0; i < n; i++)
        printf("%c:%d ", 'a' + i, niz[i]);
10    putchar('\n');
12 }

14 int main()
{
16     /* Niz u kojem ce se cuvati informacije o broju pojavljivanja malih
        slova */
        int mala_slova[26];

18

20     int c, i;

22     for(i=0;i<26;i++)

```

```

24     mala_slova[i]=0;

/* Ucitavaju se karakteri sve do kraja ulaza */
26 while((c = getchar()) != EOF)
{
28     /* Ako je procitani karakter slovo ... */
    if (isalpha(c)){
30         mala_slova[tolower(c)-'a']++;
    }
32 }

/* Ispisuju se trazene informacije */
34 ispis(mala_slova, 26);
36
38 return 0;
}

```

### Rešenje 1.1.6

```

#include <stdio.h>

2
#define MAX 100

4
int main()
{
6     int a[MAX];
    int b[MAX];

8

10     /* Rezultujuci niz ima najvise 2*MAX elemenata */
    int c[2*MAX];

12

14     int n;
    int i,j;

16

18     /* Ucitava se dimenzija nizova i proverava njena ispravnost */
    printf("Unesite dimenziju nizova:\n");
    scanf("%d", &n);
20     if (n<1 || n>MAX)
    {
22         printf("Nedozvoljena vrednost!\n");
        return -1;
24     }

26     /* Ucitavaju se elementi prvog niza */
    printf("Unesite elemente niza a:\n");
    for(i=0;i<n;i++)
    {
28         scanf("%d", &a[i]);
30     }
32

```

```

34  /* Ucitavaju se elementi drugog niza */
printf("Unesite elemente niza b:\n");
for(i=0;i<n;i++)
36  {
    scanf("%d", &b[i]);
38  }

40  /*
    Formira se treci niz.
42  Koriste se dva indeksa:

44      - indeks i pomocu kojeg se pristupa elementima nizova a i b i
        koji treba uvecati za 1 nakon svake iteracije

46      - indeks j pomocu kojeg se pristupa elementima rezultujuceg
        niza c;
48      s obzirom da se u svakoj iteraciji u niz c smestaju dva
        elementa, jedan iz niza a i jedan iz niza b,
        indeks j se uvecava za 2 nakon svake iteracije
50  */
for(i=0,j=0;i<n;i++,j+=2)
52  {
    c[j]=a[i];
54    c[j+1]=b[i];
    }

56  /* Ispisuju se elementi rezultujuceg niza */
printf("Rezultujuci niz:\n");
for(i=0;i<2*n;i++)
60    printf("%d ",c[i]);
printf("\n");
62
return 0;
64 }

```

### Rešenje 1.1.7

```

1  #include <stdio.h>

3  #define MAX 100

5  int main()
{
7    int a[MAX], b[MAX], c[2*MAX];
    int i, n;

9    /* Ucitava se broj elemenata nizova i proverava se njegova
        ispravnost */
11   printf("Unesite broj n: ");
    scanf("%d", &n);
13   if(n<1 || n>MAX)

```

```

15 {
    printf("Greska: Nedozvoljena vrednost!\n");
    return -1;
17 }

19 /* Ucitavaju se elementi nizova */
    printf("Unesite elemente niza a: ");
21 for(i=0;i<n;i++)
    scanf("%d", &a[i]);

23 printf("Unesite elemente niza b: ");
25 for(i=0;i<n;i++)
    scanf("%d", &b[i]);

27
29 /*
    Niz c ima 2*n elemenata: prvih n elemenata su elementi niza b, a
    narednih n elemenata elementi niza a.
    Elementi niza b se nalaze na pozicijama 0,1,2,...n-1, a elementi
    niza a na pozicijama
31 n,n+1,...2*n-1. Jednim prolaskom kroz petlju na poziciju i u nizu
    c se postavlja element b[i] niza b,
    a na poziciju n+i element a[i] niza a.
33 */
    for(i=0;i<n;i++)
35 {
        c[i] = b[i];
37         c[n+i] = a[i];
    }

39
41 /* Ispisuju se elementi niza c */
    for(i=0;i<2*n;i++)
        printf("%d ", c[i]);
43 printf("\n");

45 return 0;
}

```

### Rešenje 1.1.8

```

1 #include <stdio.h>

3 #define DIMENZIJA 100

5 /* Funkcija za učitavanje niza. */
void unos(int niz[], int n)
7 {
    int i;

9
    printf("Uneti elemente sortiranog niza:\n");
11 for(i=0; i<n; i++)
    scanf("%d", &niz[i]);

```



```

13 }
15 /* Funkcija za ispis niza. */
void ispis(int niz[], int n)
17 {
    int i;
19
    for(i=0; i<n; i++)
21         printf("%d ", niz[i]);

23     printf("\n");
}
25
int main()
27 {
    int a[DIMENZIJA], b[DIMENZIJA];
29     /* Niz c moze biti duplo veci od nizova a i b, pa za njega treba
        odvojiti dupli prostor. */
    int c[2*DIMENZIJA];
31     int m, n;
    /* Brojac u petlji za elemente niza a. */
33     int i = 0;
    /* Brojac u petlji za elemente niza b. */
35     int j = 0;
    /* Brojac u petlji za elemente niza c. */
37     int k = 0;

39     printf("Uneti broj elemenata niza: ");
    scanf("%d", &n);
41
    unos(a, n);
43     unos(b, n);

45     while(i<n && j<n)
    {
47         /* Porede se elementi nizova a i b i u niz c upisuje se samo onaj
            koji je manji. */
            if (a[i] < b[j])
49             {
                c[k] = a[i];
51                 /* Kako element niza a je upisan u niz c, uvecava se brojac
                    niza a.

                    Element niza b nije upisan u niz c, te brojac za niz
                    b ne treba uvecavati. */
                i++;
53             }
            else
55             {
                c[k] = b[j];
57                 j++;
59             }
    }

```

```

61     /* U nizu c na poziciju k je upisan ili a[i] ili b[j].
        Brojac k se uvecava. */
63     k++;
    }

65     /* Ukoliko je ostalo elemenata u nizu a, upisuju se u niz c. */
67     while(i < n)
    {
69         c[k] = a[i];
        k++;
71         i++;
    }

73     /* Ukoliko je ostalo elemenata u nizu b, upisuju se u niz c. */
75     while(j < n)
    {
77         c[k] = b[j];
        k++;
79         j++;
    }

81     /* Ispis elemenata niza c cija dimenzija je zbir dimenzija nizova a
        i b. */
83     ispis(c, 2*n);

85     return 0;
}

```

### Rešenje 1.1.9

```

1  #include <stdio.h>

3  #define DIMENZIJA 10

5  /* Funkcija za ucitavanje niza. */
   void unos(int niz[])
7  {
       int i;

9     printf("Unesite %d brojeva:\n", DIMENZIJA);
11    for(i=0; i<DIMENZIJA; i++)
        scanf("%d", &niz[i]);
13 }

15 /* Funkcija za ispis niza. */
   void ispis(int niz[], int n)
17 {
       int i;

19     for(i=0; i<DIMENZIJA; i++)
        printf("%d ", niz[i]);
21

```

```

23     printf("\n");
24 }
25
26 int main()
27 {
28     int niz[DIMENZIJA];
29     /* Brojaci i i j. Brojac i krece od pocetka niza, a brojac j od
        kraja. */
30     int i=0, j = DIMENZIJA-1;
31     /* Pomocna promenljiva za razmenu elemenata niza. */
32     int pom;
33
34     unos(niz);
35
36     /* Ideja u resenju je da se krene sa dva kraja niza -- sa pocetka
        niza i sa kraja i
37     svaki put kada se naidje na elemente koji po parnosti ne
        odgovaraju delu niza u kome treba da
        budu, ti elementi se zamene. */
38     while(i < j && i < DIMENZIJA && j >= 0)
39     {
40         /* Ukoliko elementi na pozicijama i i j su razlicite parnosti,
            vrsi se razmena tih elemenata niza. */
41         if (niz[i] % 2 != 0 && niz[j] % 2 == 0)
42         {
43             pom = niz[i];
44             niz[i] = niz[j];
45             niz[j] = pom;
46         }
47
48         /* Ukoliko je element na poziciji i paran, prelazi se na sledeci
            element niza,
49         brojac i se uvecava. */
50         if (niz[i] % 2 == 0)
51             i++;
52
53         /* Ukoliko je element na poziciji j neparan, prelazi se na
            sledeci element niza,
54         brojac j se smanjuje. */
55         if (niz[j] % 2 != 0)
56             j--;
57     }
58
59     printf("Rezultujuci niz:\n");
60     ispis(niz, DIMENZIJA);
61
62     return 0;
63 }
64
65

```

## Rešenje 1.1.10

```

2  #include <stdio.h>
4
6  #define MAX 100
8
10 int main()
12 {
14     int a[MAX];
16     int n;
18     int i,j;
20     char poslednja_cifra;
22     int novo_n;
24
26     /* Ucitava se dimenzija niza i proverava njena ispravnost */
28     printf("Unesite dimenziju niza:\n");
30     scanf("%d", &n);
32     if (n<1 || n>MAX)
34     {
36         printf("Nedozvoljena vrednost!\n");
38         return -1;
40     }
42
44     /* Ucitavaju se elementi niza a */
46     printf("Unesite elemente niza a:\n");
48     for(i=0;i<n;i++)
50     {
52         scanf("%d", &a[i]);
54     }
56
58     /* Obilaze se svi elementi niza a */
60     for(i=0, j=0; i<n; i++)
62     {
64         /* Izdvaja se poslednja cifra tekuceg elementa */
66         poslednja_cifra = a[i]%10;
68
69         /* Ako je poslednja cifra 0 ili je element deljiv svojom
70            poslednjom cifrom,
71            zadržavamo ga i smestamo na poziciju j */
72         if (poslednja_cifra==0 || a[i]%poslednja_cifra==0)
74         {
76             a[j]=a[i];
78             j++;
80         }
82     }
84
86     /* Dimenzija novog niza odgovara posledjoj vrednosti brojac j */
88     novo_n=j;
90
92     /* Ispisuje se rezultujuci niz */
94     printf("Niz a nakon izmena:\n");

```

```

52     for(i=0; i<novo_n;i++)
        printf("%d ", a[i]);
        printf("\n");
54
56     return 0;
}

```

### Rešenje 1.1.11

### Rešenje 1.1.12

```

#include <stdio.h>
2
#define MAX 100
4
/* a) Napisati funkciju koja ucitava elemente niza. */
6 void ucitaj(int a[], int n)
{
8     int i;
    for(i=0;i<n;i++)
10     {
        scanf("%d",&a[i]);
12     }
}
14
/* b) Napisati funkciju koja stampa elemente niza. */
16 void stampa(int a[], int n)
{
18     int i;
    for(i=0;i<n;i++)
20     printf("%d ",a[i]);
    printf("\n");
22 }
24
/* c) Napisati funkciju koja racuna sumu elemenata niza. */
int suma(int a[], int n)
26 {
    int i;
    int s=0;
    for(i=0;i<n;i++)
30     s+=a[i];
    return s;
32 }
34
/* d) Napisati funkciju koja racuna prosečnu vrednost elemenata niza.
    */
36 float prosek(int a[], int n)
{

```

```

38     int i;
39     int s = suma(a,n);
40     return (float) s/n;
41 }
42
43
44 /* e) Napisati funkciju koja izracunava maksimum elemenata niza.*/
45 int maksimum (int a[],int n)
46 {
47     int m;
48     int i;
49
50     /* Maksimum se inicijalizuje prvim elementom niza (a[0]), a zatim
51     se prolazi kroz ostatak niza.
52     U svakom koraku se poredi vrednost maksimuma sa tekucim
53     elementom niza. */
54     m = a[0];
55     for(i=1;i<n;i++)
56         if (a[i] > m)
57             m = a[i];
58
59     /* Vraca se izracunata vrednost maksimuma */
60     return m;
61 }
62
63 /* f) Napisati funkciju koja izracunava poziciju maksimalnog elementa
64    u nizu. */
65 int pozicija_maksimuma (int a[],int n)
66 {
67     int m;
68     int m_pozicija;
69     int i;
70
71     /* Minimum se inicijalizuje prvim elementom niza (a[0]) i pamti se
72     njegova pozicija (0),
73     a zatim se prolazi kroz ostatak niza.
74     U svakom koraku se poredi vrednost minimuma sa tekucim
75     elementom niza i ukoliko je potrebno menjaju se vrednosti
76     minimuma i njegove pozicije. */
77
78     m = a[0];
79     m_pozicija=0;
80     for(i=1;i<n;i++)
81         if (a[i] < m)
82         {
83             m = a[i];
84             m_pozicija=i;
85         }
86
87     /* Vraca se izracunata pozicija */
88     return m_pozicija;

```

```

84 }
86
88 int main()
89 {
90     int a[MAX];
91     int n;
92
93     /* Ucitava se dimenzija niza i proverava njena ispravnost */
94     printf("Unesite dimenziju niza:");
95     scanf("%d",&n);
96     if (n<1 || n>MAX)
97     {
98         printf("Nedozvoljena vrednost!\n");
99         return -1;
100     }
101
102     /* Testira se funkcija kojom se ucitavaju elementi niza */
103     ucitaj(a,n);
104
105     /* Testira se funkcija kojom se ispisuju elementi niza */
106     printf("Vreme trcanja takmicara: ");
107     stampaj(a,n);
108
109     /* Testira se funkcija kojom se izracunava suma elemenata niza */
110     printf("Ukupno vreme: %d\n", suma(a,n));
111
112     /* Testira se funkcija kojom se racuna prosek elemenata niza */
113     printf("Prosecno vreme trcanja: %.2f\n", prosek(a,n));
114
115     /* Testira se funkcija kojom se izracunava minimum niza */
116     printf("Maksimalno vreme trcanja: %d\n", minimum(a,n));
117
118     /* Testira se funkcija kojom se izracunava pozicija maksimalnog
119        elementa */
120     printf("Indeks pobednika: %d\n", pozicija_maksimuma(a,n));
121
122     return 0;
123 }

```

## Rešenje 1.1.13

```

1 #include <stdio.h>
2
3 #define MAX 100
4
5 /* Funkcija koja ucitava elemente niza */
6 void ucitaj(int a[], int n)
7 {

```

```

8      int i;
9      for(i=0;i<n;i++)
10     {
11         scanf("%d",&a[i]);
12     }
13 }
14
15 /* Funkcija koja ispisuje elemente niza */
16 void stampaj(int a[], int n)
17 {
18     int i;
19     for(i=0;i<n;i++)
20         printf("%d ",a[i]);
21     printf("\n");
22 }
23
24 /* a) Funkcija koja proverava da li niz sadrzi zadatu vrednost m */
25 int sadrzi(int a[], int n, int m)
26 {
27     int i;
28
29     /* Poredi se element po element niza a sa zadatim brojem m */
30
31     for(i=0;i<n;i++){
32         /* Ukoliko je tekuci element niza jednak trazenom broju */
33         if (a[i]==m){
34             /* Funkcija vraca vrednost 1 */
35             return 1;
36         }
37     }
38
39     /* Ako se stigne do kraja niza i ne naidje na vrednost koja je
40        jednaka broju m,
41        to znaci da se broj ne nalazi u nizu i da funkcija treba da
42        vrati 0. */
43     return 0;
44 }
45
46 /* b) Funkcija koja vraca vrednost prve pozicije na kojoj se nalazi
47     element koji ima vrednost m,
48     ili -1 ukoliko element nije u nizu */
49 int prvo_pojavljivanje(int a[], int n, int m)
50 {
51     int i;
52
53     /* Poredi se element po element niza a sa zadatim brojem m */
54
55     for(i=0;i<n;i++){
56         /* Ukoliko je tekuci element niza jednak trazenom broju */
57         if (a[i]==m){
58             /* Vraca se njegov indeks */

```



```

58     return i;
59 }
60 }
61
62 /* Ako se stigne do kraja niza i ne naidje na vrednost koja je
63    jednaka broju m,
64    to znaci da se broj ne nalazi u nizu i da funkcija treba da
65    vrati -1. */
66 return -1;
67 }
68
69 /* c) Funkcija koja vraca vrednost poslednje pozicije na kojoj se
70    nalazi element koji ima vrednost m,
71    ili -1 ukoliko element nije u nizu */
72 int poslednje_pojavljivanje(int a[], int n, int m)
73 {
74     int i;
75
76     /* Polazi se od kraja niza i poredi se element po element sa
77        zadatim brojem m*/
78
79     for(i=n-1;i>=0;i--){
80         /* Ukoliko je tekuci element niza jednak trazenom broju */
81         if (a[i]==m){
82             /* Vraca se njegov indeks */
83             return i;
84         }
85     }
86
87     /* Ako se stigne do pocetka niza i ne naidje na vrednost koja je
88        jednaka broju m,
89        to znaci da se broj ne nalazi u nizu i da funkcija treba da
90        vrati -1. */
91     return -1;
92 }
93
94 int main()
95 {
96     int a[MAX];
97     int n;
98     int m;
99     int i;
100
101     /* Ucitava se dimenzija niza i proverava se njena ispravnost */
102     printf("Unesite dimenziju niza:");
103     scanf("%d",&n);
104     if (n<1 || n>MAX)
105     {
106         printf("Nedozvoljena vrednost!\n");
107         return -1;
108     }
109 }

```

```

104      /* Ucitavaju se i ispisuju elementi niza */
      ucitaj(a,n);
106      printf("Ucitani niz:");
      stampaj(a,n);

108
      /* Ucitava se vrednost za pretragu */
110      printf("Unesi jedan ceo broj:");
      scanf("%d",&m);

112
      /* I proverava se rad napisanih funkcija */
114      if(sadrzi(a,n,m))
          printf("Niz sadrzi element cija je vrednost %d\n", m);
116      else
          printf("Niz ne sadrzi element cija je vrednost %d\n", m);
118
      i = prvo_pojavljivanje(a,n,m);
120      if(i!=-1)
          printf("Indeks njegovog prvog pojavljivanja u nizu je %d\n", m,
              i);
122
      i = poslednje_pojavljivanje(a,n,m);
124      if(i!=-1)
          printf("Indeks njegovog poslednjeg pojavljivanja u nizu je %d\n",
              m,i);
126
128      return 0;
  }

```

### Rešenje 1.1.14

```

#include<stdio.h>

2
#define MAX 100

4
/* Funkcija kojom se ucitavaju elementi niza a dimenzije n */
6 void ucitaj(int a[], int n)
{
8     int i;
    for(i=0;i<n;i++)
10     {
        scanf("%d",&a[i]);
12     }
}

14
/* Funkcija kojom se ispisuju elementi niza a dimenzije n */
16 void stampaj(int a[], int n)
{
18     int i;
    for(i=0;i<n;i++)

```

```

20         printf("%d ",a[i]);
21     printf("\n");
22 }
23
24
25
26 /* b) Funkcija koja obrce elemente niza */
27 void obrni(int a[], int n)
28 {
29
30     int t;
31     int i,j;
32
33
34     /*
35      Za niz a[0], a[1], ..., a[n-2], a[n-1] obrnuti niz je a[n-1], a[
36      n-2], ..., a[1], a[0]
37      Zato je potrebno razmeniti vrednosti elemenata a[0] i a[n-1], a
38      [1] i a[n-2], itd. i zaustaviti se
39      kada je vrednost indeksa prvog elementa veca od vrednosti drugog
40      elementa.
41     */
42
43     for(i=0,j=n-1;i<j;i++, j--)
44     {
45         t = a[i];
46         a[i] = a[j];
47         a[j] = t;
48     }
49 }
50
51 /* c) Funkcija koja rotira niz ciklicno za jedno mesto u levo */
52 void rotiraj1(int a[], int n)
53 {
54     int i;
55     int tmp;
56
57     /* Izdvaja se prvi element niza */
58     tmp=a[0];
59
60     /* Pomeraju se preostali elementi niza */
61     for(i=0;i<n-1;i++){
62         a[i]=a[i+1];
63     }
64
65     /* Poslednjem elementu se dodeljuje sacuvana vrednost prvog
66     elementa */
67     a[n-1] = tmp;
68 }
69
70 /* d) Funkcija koja rotira niz ciklicno za k mesta u levo */

```

```

68 void rotirajk(int a[], int n, int k)
69 {
70     int i;
71
72     /* Odredjuje se vrednost broja k koja je u opsegu od 0 do n-1 kako
       bi se izbegla suvisna pomeranja */
       k=k%n;
74
75     /* Niz se rotira za jednu poziciju ulevo k puta */
76     for(i=0;i<k;i++)
77         rotiraj1(a,n);
78 }
79
80 int main()
81 {
82     int a[MAX];
83     int n;
84     int i;
85     int k;
86     int m;
87
88     /* Ucitava se dimenzija niza i proverava se njena ispravnost */
89     printf("Unesite dimenziju niza:");
90     scanf("%d",&n);
91     if (n<1 || n>MAX)
92     {
93         printf("Nedozvoljena vrednost!\n");
94         return -1;
95     }
96
97     /* Ucitavaju se elementi niza */
98     ucitaj(a,n);
99
100
101     /* Testira se rad napisanih funkcija */
102
103     /* a) */
104     printf("Elementi niza nakon obrtanja:\n");
105     obrni(a,n);
106     stampaj(a,n);
107
108     /* b) */
109     printf("Elementi niza nakon rotiranja za 1 mesto ulevo:\n");
110     rotiraj1(a,n);
111     stampaj(a,n);
112
113     /* c) */
114     printf("Unesite jedan pozitivan ceo broj:");
115     scanf("%d",&k);
116     if (k<=0)
117     {
118         printf("Nekorektan unos\n");

```

```

        return -1;
120     }
    rotirajk(a,n,k);
122     printf("Elementi niza nakon rotiranja za %d mesto ulevo:\n",k);
    stampaj(a,n);
124
126     return 0;
}

```

## Rešenje 1.1.15

```

1  #include <stdio.h>
3  #define MAX 100
5  int main()
6  {
7      float brojevi[MAX];
8      int n, i;
9
10     /* Ucitava se dimenzija niza i proverava se njena ispravnost */
11     printf("Unesite broj elemenata niza: ");
12     scanf("%d", &n);
13     if(n<1 || n>MAX)
14     {
15         printf("Nedozvoljena vrednost!\n");
16         return -1;
17     }
18
19     /* Ucitavaju se elementi niza */
20     printf("Unesite elemente niza:\n");
21     for(i=0;i<n;i++){
22         scanf("%f", &brojevi[i]);
23     }
24
25     /* Ukoliko je i-ti element niza brojevi[i] negativan broj,
26        kvadriramo ga tako sto ga pomnozimo sa samim sobom. */
27     for(i=0;i<n;i++){
28         if(brojevi[i]<0)
29             brojevi[i] *= brojevi[i];
30     }
31
32     /* Ispisuje se novodobijeni niz */
33     for(i=0;i<n;i++){
34         printf("%g ", brojevi[i]);
35     }
36     printf("\n");
37
38     return 0;
39 }

```

## Rešenje 1.1.16

```
1 #include <stdio.h>
3 #define MAX 100
5 int main()
6 {
7     int brojevi[MAX];
8     int n, i, k, indikator;
9
10    /* Ucitava se dimenzija niza i proverava se njena ispravnost */
11    printf("Unesite dimenziju niza: ");
12    scanf("%d", &n);
13    if(n<1 || n>MAX)
14    {
15        printf("Greska: Nedozvoljena vrednost!\n");
16        return -1;
17    }
18
19    /* Ucitavaju se elementi niza */
20    printf("Unesite elemente niza: ");
21    for(i=0; i<n; i++)
22        scanf("%d", &brojevi[i]);
23
24
25    /* Ucitava se broj k i proverava se njegova ispravnost */
26    printf("Unesite broj k: ");
27    scanf("%d", &k);
28    if(k == 0)
29    {
30        printf("Greska: Pogresan unos!\n");
31        return -1;
32    }
33
34    /*
35     Promenljiva koja cuva informaciju o tome da li je u nizu
36     postojao element koji je deljiv brojem k.
37     Inicijalna vrednost je 0.
38     */
39    indikator = 0;
40
41    /*
42     Ukoliko je element niza deljiv brojem k, indikator se postavlja
43     na 1
44     i ispisuje se indeks tog elementa.
45     */
```

```

47     for(i=0;i<n;i++){
48         if(brojevi[i]%k == 0)
49         {
50             indikator = 1;
51             printf("%d ",i);
52         }
53     }
54
55     /* Ukoliko je indikator jednak nuli to znaci da ne postoji element
56        u nizu koji je deljiv brojem k. */
57
58     if(indikator == 0){
59         printf("U nizu nema elemenata koji su deljivi brojem %d!\n",k);
60     }
61
62     return 0;
63 }

```

## Rešenje 1.1.17

```

1  #include <stdio.h>
2
3  #define MAX 100
4
5  int main()
6  {
7      int brojevi[MAX];
8      int n, i, poz_max, poz_min, max, min, tmp;
9
10     /* Ucitava se dimenzija niza i proverava se njena ispravnost */
11     printf("Unesite dimenziju niza: ");
12     scanf("%d", &n);
13     if(n<1 || n>MAX)
14     {
15         printf("Greska: Nedozvoljena vrednost!\n");
16         return -1;
17     }
18
19     /* Ucitavaju se elementi niza */
20     printf("Unesite elemente niza:\n");
21     for(i=0;i<n;i++){
22         scanf("%d", &brojevi[i]);
23     }
24
25     /*
26      Maksimalnim tj. minimalnim elementom niza proglašava se nulti
27      element niza.
28      Pozicije maksimalnog tj. minimalnog elementa se postavljaju na 0.
29     */
30     max = brojevi[0];
31     min = brojevi[0];

```

```

31     poz_max = 0;
    poz_min = 0;

33     /* U prolazu kroz niz trazi se maksimalni i minimalni element i
        pamte se njihove pozicije */
    for(i=1;i<n;i++)
35     {
        if(brojevi[i] > max)
37         {
            max = brojevi[i];
            poz_max = i;
39         }

41         if(brojevi[i] < min)
43         {
            min = brojevi[i];
            poz_min = i;
45         }
47     }

49     /* Zamenjuju se elementi na pozicijama poz_min i poz_max */
    tmp = max;
    brojevi[poz_max] = min;
    brojevi[poz_min] = tmp;

53     /* Ispisuje se rezultujuci niz */
    for(i=0;i<n;i++)
55         printf("%d ", brojevi[i]);
57     printf("\n");

59     return 0;
61 }

```

## Rešenje 1.1.18

## Rešenje 1.1.19

```

1  #include <stdio.h>

3  #define MAX 100

5  int main()
6  {
7      /* niz karaktera */
8      char karakteri[MAX];
9      char c;
10     int i, n;
11

```



```

13  for(i=0;i<MAX;i++)
14  {
15      /* Ucitava se karakter po karakter sa standardnog ulaza sve dok
16      se ne unese * ili
17      se ne prekorači maksimalni broj karaktera */
18      printf("Unesite karakter: ");
19      scanf("%c", &c);
20
21      /* Cita se znak za novi red nakon unesenog karaktera */
22      getchar();
23
24      /* Ukoliko je unet karakter * prekida se dalje citanje i izlazi
25      se iz petlje */
26      if(c == '\n')
27          break;
28
29      /* Inace, procitani karakter se smesta u niz */
30      karakteri[i] = c;
31  }
32
33  /* Broj unetih karaktera je nakon izlaska iz petlje i-1 */
34  n = i-1;
35
36  /* Ispisuju se karakteri u obrnutom redosledu */
37  for(i=n;i>=0;i--)
38  {
39      printf("%c ", karakteri[i]);
40  }
41  printf("\n");
42
43  return 0;
44  }

```

## Rešenje 1.1.20

```

1  #include <stdio.h>
2
3  #define MAX 100
4
5  /* Funkcija koja vraca broj pojavljivanja broja x u nizu */
6  int broj_pojavljivanja(int niz[], int n, int x)
7  {
8      int i;
9
10     /* Broj pojavljivanja broja x */
11     int brojac = 0;
12
13     /* Obilazi se element po element niza */
14     for(i=0;i<n;i++){
15         /* Ukoliko je tekuci element jednak traženom broju */

```

```

17     if(niz[i] == x){
18         /* Uvecava se broj pojavljivanja */
19         brojac++;
20     }
21 }
22
23 /* Vraca se izracunata vrednost */
24 return rezultat;
25 }
26
27 int main()
28 {
29     /* Niz elemenata koje zadaje korisnik */
30     int a[MAX];
31
32     /* Niz elemenata koji se pojavljuju tri puta */
33     int b[MAX];
34
35     int i, j, n, n_b;
36
37     /* Ucitava se broj elemenata korisnickog niza i proverava se
38        njegova ispravnost */
39     printf("Unesite broj n: ");
40     scanf("%d", &n);
41     if(n<1 || n>MAX)
42     {
43         printf("Greska: Nedozvoljena vrednost!\n");
44         return -1;
45     }
46
47     /* Ucitavaju se elementi korisnickog niza */
48     printf("Unesite elemente niza a: ");
49     for(i=0;i<n;i++)
50         scanf("%d", &a[i]);
51
52     /* j - brojac elemenata rezultujuceg niza b */
53     j = 0;
54
55     /* Obilazi se element po element niza a */
56     for(i=0;i<n;i++)
57     {
58         /* Ukoliko se tekuci element pojavljuje tacno tri puta u nizu a i
59            nije upisan u niz b
60            koji trenutno ima j elemenata, dodaje se u niz b na poziciju j i
61            uvecava se broj elemenata niza b */
62         if(broj_pojavljivanja(a, n, a[i])==3 && broj_pojavljivanja(b, j,
63            a[i])==0)
64         {
65             b[j] = a[i];
66             j++;
67         }
68     }
69 }

```

```

65  /* Ispisuje se rezultujući niz b - broj elemenata u nizu b je j*/
    n_b = j;
67  for(i=0;i<n_b;i++)
        printf("%d ", b[i]);
69  printf("\n");

71  return 0;
}

```

Rešenje 1.1.21

Rešenje 1.1.22

```

#include <stdio.h>

2
#define MAX 100

4
/*
6  Funkcija koja vraća 1 ukoliko broj x postoji u nizu, 0 inace.
*/

8
int postoji(int niz[], int n, int x)
10 {
    int i;
12
    for(i=0;i<n;i++)
14         if(niz[i] == x)
                return 1;

16     return 0;
18 }

20 int main()
{
22     int a[MAX], b[MAX], unija[2*MAX], presek[MAX], razlika[MAX];
    int i, j, n_a, n_b, n_u, n_p, n_r, indikator;

24
    printf("Unesite broj elemenata niza a: ");
26     scanf("%d", &n_a);

28     if(n_a<1 || n_a>100)
    {
30         printf("Greska: pogresan unos!\n");
        return -1;
32     }

34     printf("Unesite elemente niza a: ");
    for(i=0;i<n_a;i++)
36         scanf("%d", &a[i]);

```

```

38 printf("Unesite broj elemenata niza b: ");
   scanf("%d", &n_b);

40
42 if(n_b<1 || n_b>100)
   {
44     printf("Greska: pogresan unos!\n");
       return -1;
   }

46
48 printf("Unesite elemente niza b: ");
   for(i=0;i<n_b;i++)
       scanf("%d", &b[i]);

50
52 /*
   Brojaci elemenata u nizovima unija, presek i razlika.
   */
54 n_u = 0;
   n_p = 0;
56 n_r = 0;

58 for(i=0;i<n_a;i++)
   {
60     /*
        Ukoliko se element a[i] ne nalazi u uniji, dodajemo ga u uniju
        i povecamo brojac elemenata u nizu unija.
        */
62     if(postoji(unija,n_u,a[i]) == 0)
64     {
        unija[n_u] = a[i];
66         n_u++;
        }

68
70     /*
        Ukoliko se element a[i] postoji u nizu b i ne postoji u nizu
        presek, dodajemo ga u presek i povecavamo brojac elemenata u nizu
        presek.
        */
72     if(postoji(b, n_b, a[i])==1 && postoji(presek, n_p, a[i])==0)
74     {
        presek[n_p] = a[i];
        n_p++;
76     }

78
80     /*
        Ukoliko element a[i] ne postoji u nizu b i ne postoji u nizu
        razlika, dodajemo ga u razliku i povecavamo brojac elemenata u
        nizu razlika.
        */
82     if(postoji(b, n_b, a[i])==0 && postoji(razlika, n_r, a[i])==0)
        {
            razlika[n_r] = a[i];

```

```

84         n_r++;
85     }
86 }
87
88 /*
89  Elemente niza b koji ne postoje u uniji dodajemo u uniju.
90 */
91 for(i=0;i<n_b;i++)
92     if(postoji(unija, n_u, b[i]))
93     {
94         unija[n_u] = b[i];
95         n_u++;
96     }
97
98 printf("Unija: ");
99 for(i=0;i<n_u;i++)
100     printf("%d ", unija[i]);
101
102 printf("\nPresek: ");
103 for(i=0;i<n_p;i++)
104     printf("%d ", presek[i]);
105
106 printf("\nRazlika: ");
107 for(i=0;i<n_r;i++)
108     printf("%d ", razlika[i]);
109
110 return 0;
111 }

```

### Rešenje 1.1.23

```

1  #include <stdio.h>
2
3  #define MAX 100
4
5  int main()
6  {
7      int a[MAX];
8      int i, j, n_a;
9
10     /* Ucitava se broj elemenata niza i proverava se njegova ispravnost
11      */
12     printf("Unesite broj elemenata niza: ");
13     scanf("%d", &n_a);
14     if(n_a<1 || n_a>100)
15     {
16         printf("Greska: Nedozvoljena vrednost!\n");
17         return -1;
18     }
19
20     /* Ucitavaju se elementi niza */

```

```

20 printf("Unesite elemente niza: ");
21 for(i=0;i<n_a;i++)
22     scanf("%d", &a[i]);
23
24 /*
25 1. nacin
26
27 int b[MAX], n_b;
28
29 // Brojac j predstavlja poziciju u nizu b na koju treba smestiti
30 element niza a. Njegova pocetna vrednost je 0.
31
32 for(i=0, j=0;i<n_a;i++){
33     // Ako je tekuci element niza a paran
34     if(a[i]%2 == 0)
35     {
36         // Smesta se na poziciju j u nizu b
37         b[j] = a[i];
38         // Vrednost brojaca j se priprema za narednu iteraciju
39         j++;
40     }
41
42     // Ako je element niza a neparan, sa njim nista ne treba raditi
43 }
44
45 // Broj elemenata novodobijenog niza b je j
46 n_b = j;
47
48 // Ispisuju se elementi niza b
49 for(i=0;i<n_b;i++)
50     printf("%d ", b[i]);
51
52 */
53
54 /*
55 2. nacin
56
57 J predstavlja brojac prve slobodne pozicije na koju se moze
58 upisati element niza koji treba da ostane u nizu.
59 Kada se naidje na element koji je paran, on se kopira na mesto a[j]
60 i poveca se vrednost brojaca j.
61 Ukoliko se naidje na element koji je neparan, njega treba
62 preskociti.
63 */
64 for(i=0, j=0;i<n_a;i++)
65 {
66     /* Ako je tekuci element niza a paran */
67     if(a[i]%2 == 0)
68     {
69         /* Premesta se na poziciju j */
70         a[j] = a[i];

```

```

68     /* Vrednost brojaca j se priprema za narednu iteraciju */
70     j++;
72 }
74     /* Ako je tekuci element niza a neparan, sa njim nista ne treba
       raditi */
76 }
78     /* U nizu a se sada na pozicijama od 0,...,j-1 nalaze elementi koji
       su parni, te je njegova nova dimenzija j. */
80     n_a=j;
82     /* Ispisuju se elementi modifikovanog niza a */
84     for(i=0;i<n_a;i++){
86         printf("%d ", a[i]);
87     }
88     printf("\n");
89     return 0;
90 }

```

## Rešenje 1.1.24

```

1 #include <stdio.h>
2 #include <math.h>
3
4 #define MAX 100
5
6 /*
7  Funkcija koja proverava da li je zadati broj prost broj.
8  Povratna vrednost funkcije je 1 ukoliko broj jeste prost, inace je
9  0.
10 */
11 int prost(int x)
12 {
13     int i;
14
15     /* Posmatra se apsolutna vrednost broja kako bi se pokrio i slucaj
       negativnih brojeva */
16     x=abs(x);
17
18     /* Brojevi 1, 2 i 3 su prosti */
19     if(x == 1 || x == 2 || x == 3)
20         return 1;
21
22     /* Ako je broj paran nije prost */
23     if(x%2 == 0)
24         return 0;
25
26     for(i=3;i<=sqrt(x);i+=2){
27         if(x%i==0)
28             return 0;
29     }
30     return 1;
31 }

```

```

/* Ako broj ima delioce u skupu [3, koren_broja(x)] takodje nije
   prost */
26 for(i=3;i<=sqrt(x);i+=2){
    if(x%i == 0)
28     return 0;
}

/* Ako su svi uslovi ispunjeni, broj je prost */
32 return 1;
}

34
int main()
36 {
    int a[MAX];
38     int i, j, n_a, n_b;

40     /* Ucitava se broj elemenata niza i proverava se njegova ispravnost
       */
    printf("Unesite broj elemenata niza: ");
42     scanf("%d", &n_a);
    if(n_a<1 || n_a>MAX)
44     {
        printf("Greska: Nedozvoljena vrednost!\n");
46         return -1;
    }

48
    /* Ucitavaju se elementi niza a */
50     printf("Unesite elemente niza: ");
    for(i=0;i<n_a;i++)
52         scanf("%d", &a[i]);

54
    /*
56     1. nacin

58     int b[MAX];

60

62     for(i=0, j=0;i<n_a;i++){
        if(prost(a[i]) == 0)
        {
64             b[j] = a[i];
            j++;
66         }
    }

68
    // Broj elemenata novodobijenog niza b je j
70     n_b = j;

72     // Ispisuju se elementi niza b
    for(i=0;i<n_b;i++)
74         printf("%d ", b[i]);

```



```

76     printf("\n");
78
79     */
80     /*
81     2. nacin
82     */
83
84     for(i=0, j=0; i<n_a; i++)
85     {
86         if(prost(a[i]) == 0)
87         {
88             a[j] = a[i];
89             j++;
90         }
91     }
92
93     n_a = j;
94
95     /* Ispisuju se elementi modifikovanog niza a */
96     for(i=0; i<n_a; i++)
97         printf("%d ", a[i]);
98     printf("\n");
99
100    return 0;
101}

```

## Rešenje 1.1.25

```

1  #include <stdio.h>
2
3  #define MAX 100
4
5  /* Funkcija prebrojavanje vraca broj elemenata niza koji su manji od
6     poslednjeg elementa*/
7  int prebrojavanje(int a[], int n)
8  {
9
10     int i;
11
12     /* Brojac elemenata koji su manji od poslednjeg */
13     int broj_manjih=0;
14
15     /* Obilazi se element po element niza */
16     for(i=0; i<n-1; i++){
17         /* Ako je tekuci element manji od poslednjeg (on se nalazi na
18            poziciji n-1) */
19         if(a[i]<a[n-1]){
20             /* Uvecava se brojac */
21             broj_manjih++;
22         }
23     }
24 }

```

```

22  /* Vraca se izracunata vrednost */
    return broj_manjih;
24 }

26 int main()
{
28     int a[MAX];
    int n;
30     int i;

32     /* Ucitava se broj elemenata niza i proverava se njegova ispravnost
        */
    printf("Unesite broj elemenata niza:");
34     scanf("%d", &n);
    if(n<=0 || n>MAX)
36     {
        printf("Greska: Nedoovoljena vrednost!\n");
38         return 0;
    }

40     /* Ucitavaju se elementi niza */
    printf("Unesite elemente niza:");
42     for(i=0;i<n;i++)
        scanf("%d",&a[i]);
44

46     /* Ispisuje se rezultat poziva funkcije */
    printf("%d\n", prebrojavanje(a,n));
48
    return 0;
50 }

```

### Rešenje 1.1.26

```

#include <stdio.h>

2
#define MAX 100

4
/* Funkcija vraca broj parnih elemenata niza koji prethode
    maksimalnom elementu niza */
6 int prebrojavanje(int a[], int n)
{
8     int i;

10     int maksimum;
    int pozicija_maksimuma;

12

14     /* Brojac elemenata koji su parni i prethode maksimalnom */
    int broj_parnih;

16     /* Pronalazi se maksimalni element niza i njegova pozicija */

```

```

18     maksimum = a[0];
    pozicija_maksimuma = 0;

20     for(i=1; i<n-1; i++)
        if(a[i] > maksimum)
22         {
            maksimum = a[i];
24             pozicija_maksimuma = i;
        }

26     /* Prebrojavaju se parni elementi koji prethode maksimalnom */
28     broj_parnih = 0;
    for(i=0; i < pozicija_maksimuma; i++){
30         if(a[i]%2==0){
            broj_parnih++;
32         }
    }

34     /* Vraca se izracunata vrednost */
36     return broj_parnih;
}

38 int main()
40 {
    int a[MAX];
42     int n;
    int i;

44     /* Ucitava se broj elemenata niza i proverava se njegova ispravnost
    */
46     printf("Unesite broj elemenata niza:");
    scanf("%d", &n);
48     if(n<=0 || n>MAX)
    {
50         printf("Greska: Nedozvoljena vrednost!\n");
        return 0;
52     }

54     /* Ucitavaju se elementi niza */
    printf("Unesite elemente niza:");
56     for(i=0; i<n; i++){
        scanf("%d", &a[i]);
58     }

60     /* Ispisuje se rezultat poziva funkcije */
    printf("%d\n", prebrojavanje(a,n));

62     return 0;
64 }

```

Rešenje 1.1.27

```

1  #include <stdio.h>
   #include <ctype.h>
3
5  #define MAX 100
7  /* Funkcija prebrojava cifre u datom nizu karaktera */
   int cifre(char a[], int n)
   {
9     int i;

11    /* Brojac cifara */
    int broj_cifara = 0;

13    /* Obilazi se element po element niza */
    for(i=0; i<n; i++){
15        /* Ako je tekuci element cifra */
        if(isdigit(a[i])){
17            /* Uvecava se broj cifara */
            broj_cifara++;
19        }
21    }

23    /* Vraca se izracunata vrednost */
    return broj_cifara;
25 }

27 int main()
   {
29     char a[MAX];
    int n;
31     int i;

33     /* Ucitava se broj elemenata niza i proverava se njegova ispravnost
       */
    printf("Unesite broj elemenata niza:");
35     scanf("%d", &n);
    if(n<=0 || n>MAX)
37     {
        printf("Greska: Nedozvoljena vrednost!\n");
39         return 0;
    }

41    /* Ucitavaju se elementi niza */
    printf("Unesite elemente niza:");
43     for(i=0; i<n; i++) {
45         /* Preskace se prethodno uneti znak za novi red */
        getchar();

47         /* A zatim se ucitava sam karakter i smesta u niz */
49         scanf("%c", &a[i]);
    }

```

```

51     /* Ispisuje se rezultat poziva funkcije */
53     printf("Broj cifara je: %d\n", cifre(a,n));

55     return 0;
}

```

## Rešenje 1.1.28

```

1  #include<stdio.h>

3  #define MAX 100

5  /* Funkcija racuna zbir elemenata niza od pozicije i do pozicije j */
   int zbir(int a[], int n, int i, int j){
7      int k;

9      /* Zbir elemenata niza iz zadatog opsega */
      int z = 0;

11     /* Obilaze se elementi niza */
13     for(k=i; k<=j; k++){
         z+=a[k];
15     }

17     /* Vraca se izracunata vrednost */
      return z;
19 }

21 int main(){

23     int n, i, j;
      int a[MAX];

25     /* Ucitava se broj elemenata niza i proverava se njegova ispravnost
       */
27     printf("Unesite broj elemenata niza: ");
      scanf("%d", &n);
29     if(n <= 0 || n > MAX)
      {
31         printf("Greska: Nedozvoljena vrednost!\n");
         return 0;
33     }

35     /* Ucitavaju se elementi niza */
      printf("Unesite elemente niza:");
37     for(i=0; i<n; i++)
         scanf("%d", &a[i]);

39     /* Ucitavaju se vrednosti granica */
41     printf("Unesite vrednosti za i i j: ");

```

```

scanf("%d%d", &i, &j);

43
/* Proverava se korektnost zadatog intervala */
45 if(i < 0 || j < 0 || i > n-1 || j > n-1 || i > j){
printf("Greska: Nekorektne vrednosti granica!\n");
47 return 0;
}

49
/* Ispisuje se rezultat poziva funkcije */
51 printf("Zbir je: %d", zbir(a,n,i,j));

53 return 0;
}

```

## Rešenje 1.1.29

```

#include<stdio.h>

2
#define MAX 100

4
/* Funkcija racuna zbir prvih k pozitivnih elemenata niza */
6 float zbir_pozitivnih(float a[], int n, int k){

8     int i;

10     /* Zbir pozitivnih elemenata */
float zbir=0;

12
/* Obilazi se element po element niza - postupk se zavrsava ukoliko
se dodje do kraja niza
ili ukoliko se sabere k pozitivnih elemenata */
14 for(i=0; i<n && k>0; i++){
16     /* Ako je tekuci element pozitivan broj */
if(a[i] >= 0){
18         /* Dodaje se zbiru */
zbir+=a[i];
20         /* I umanjuje se brojac pozitivnih elemenata */
k--;
22     }
}

24
/* Vraca se izracunata vrednost */
26 return zbir;
}

28
int main(){
30     int n, i, k;
float a[MAX];

32
/* Ucitava se broj elemenata niza i proverava se njegova ispravnost
*/

```

```

34 printf("Unesite broj elemenata niza: ");
   scanf("%d", &n);
36 if(n<=0 || n> MAX){
   printf("Greska: Nedozvoljena vrednost!\n");
38   return 0;
   }

40
   /* Ucitavaju se elementi niza */
42 printf("Unesite elemente niza: ");
   for(i=0; i<n; i++)
44   scanf("%f", &a[i]);

46 /* Ucitava se broj k i proverava se njegova ispravnost */
   printf("Unesite vrednost za k: ");
48   scanf("%d", &k);
   if(k<0 || k>n){
50     printf("Greska: Nedozvoljena vrednost!");
     return 0;
52   }

54 /* Ispisuje se rezultat poziva funkcije */
   printf("Zbir je: %.2f\n", zbir_pozitivnih(a,n,k));
56   return 0;
58 }

```

### Rešenje 1.1.30

```

1  #include<stdio.h>

3  #define MAX 100

5  /* Funkcija koja kvadrira elemente niza koji se nalaze na parnim
   pozicijama */
   void kvadriranje(float a[], int n){
7
   int i;
9
   /* Obilaze se elementi na parnim pozicijama */
11  for(i=0; i<n; i+=2){
     /* I kvadriraju se: a[i] = a[i]*a[i] */
13     a[i]*=a[i];
   }
15 }

17
19 int main(){
21   int n, i, j;
   float a[MAX];

```

```

23  /* Ucitava se broj elemenata niza i proverava se njegova ispravnost
    */
    printf("Unesite broj elemenata niza: ");
25  scanf("%d", &n);
    if(n <=0 || n>MAX)
27  {
        printf("Greska: Nedozvoljena vrednost!\n");
29  return 0;
    }

31
    /* Ucitavaju se elementi niza */
33  printf("Unesite elemente niza:");
    for(i=0; i<n; i++)
35  scanf("%f", &a[i]);

37  /* Poziva se funkcije */
    kvadriranje(a,n);
39

    /* Ispisuje se elementi novodobijenog niza */
41  /* Koriscenje specifikatora %g za stampanje realnih brojeva
    omogucava ispis broja
    na onoliko decimalnih mesta koliko ima i sam broj */
43  for(i=0; i<n; i++){
        printf("%g ", a[i]);
45  }
        printf("\n");
47
    return 0;
49 }

```

Rešenje [1.1.31](#)

Rešenje [1.1.32](#)

Rešenje [1.1.33](#)

Rešenje [1.1.34](#)

Rešenje [1.1.35](#)

**Zadatak 1.2.1** Napisati program koji za dva cela broja  $x$  i  $y$  koja se učitavaju sa standardnog ulaza proverava da li se zapisuju pomoću istih cifara. UPUTSTVO: Rešiti korišćenjem nizova. Pogledati zadatak ??.



*Primer 1*

```
|| INTERAKCIJA SA PROGRAMOM:  
| Unesite dva broja: 251 125  
| Brojevi se zapisuju istim ciframa!
```

*Primer 2*

```
|| INTERAKCIJA SA PROGRAMOM:  
| Unesite dva broja: 8898 9988  
| Brojevi se ne zapisuju istim ciframa!
```

*Primer 3*

```
|| INTERAKCIJA SA PROGRAMOM:  
| Unesite dva broja: -7391 1397  
| Brojevi se zapisuju istim ciframa!
```

[Rešenje 1.2.1]