

## PROGRAMIRANJE 1



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

# **PROGRAMIRANJE 1**

## **Zbirka zadataka**

**Beograd  
2016.**

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: Nedoovoljena 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 za dva cela broja  $x$  i  $y$  koja se učitavaju sa standardnog ulaza proverava da li se zapisuju pomoću istih cifara.

*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 1.1.4]

**Zadatak 1.1.5** 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
```

[Rešenje 1.1.5]

**Zadatak 1.1.6** 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:
haHIjkl
a:1 b:0 c:0 d:0 e:0 f:0 g:0 h:2 i:0 j:2 k:1 l:1 m:0
n:0 o:0 p:0 q:0 r:0 s:0 t:0 u:0 v:0 w:0 x:0 y:0 z:0
```

### Primer 2

```
INTERAKCIJA SA PROGRAMOM:
DanaS j3 _j_utRo laBU78d
a:1 b:0 c:0 d:0 e:0 f:0 g:0 h:2 i:0 j:2 k:1 l:1 m:0
n:0 o:0 p:0 q:0 r:0 s:0 t:0 u:0 v:0 w:0 x:0 y:0 z:0
```

### Primer 3

```
INTERAKCIJA SA PROGRAMOM:
Sao Paolo 1998 _JuZna Amerika90
a:5 b:0 c:0 d:2 e:1 f:0 g:0 h:0 i:1 j:1 k:1 l:1 m:1
n:1 o:3 p:1 q:0 r:1 s:1 t:0 u:1 v:0 w:0 x:0 y:0 z:0
```

**Zadatak 1.1.7** 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 1.1.7]

**Zadatak 1.1.8** Sa standardnog ulaza se učitava ceo broj  $n$  (manji od 100) i elementi dva niza  $a$  i  $b$  dimenzije  $n$ . Napisati program koji formira i ispisuje niz  $c$  čiju prvu polovinu čine elementi niza  $b$ , a drugu polovinu elementi niza  $a$ .

### 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 1.1.8]

**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:
-2 8 20 -8 14 3 17 59 53 11
```

*Primer 2*

```
INTERAKCIJA SA PROGRAMOM:
Unesite 10 brojeva:
9 142 -9 -278 -69 33 34 28 -6 14
Rezultujući niz:
142 -278 34 28 -6 14 33 -69 -9 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!
```

[Rešenje 1.1.10]

**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

Napisati funkcije za rad sa nizovima celih brojeva.

- (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 sumu elemenata niza  $a$  dimenzije  $n$ .
- (d) Napisati funkciju `int prosek(int a[], int n)` koja računa i vraća prosečnu vrednost (aritmetičku sredinu) elemenata niza  $a$  dimenzije  $n$ .
- (e) Napisati funkciju `int minimum(int a[], int n)` koja izračunava i vraća minimum elemenata niza  $a$  dimenzije  $n$ .
- (f) Napisati funkciju `int pozicija_maksimuma(int a[], int n)` koja izračunava i vraća poziciju maksimalnog elementa u nizu  $a$  dimenzije  $n$ . U slučaju više pojavljivanja maksimalnog elementa, vratiti najmanju poziciju.

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:
5
2 5 -2 8 11
Ucitani niz: 2 5 -2 8 11
Suma elemenata niza: 24
Prosečna vrednost elemenata niza: 4.80
Minimumalni element niza: -2
Indeks maksimalnog elementa niza: 4
```

#### Primer 2

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

[Rešenje 1.1.12]

### Zadatak 1.1.13

Napisati funkcije za rad sa nizovima celih brojeva.

- (a) Napisati funkciju koja proverava da li niz sadrži zadatu vrednost  $m$ . 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 vrednost  $m$  ili  $-1$  ukoliko element nije u nizu.

- (c) Napisati funkciju koja vraća vrednost poslednje pozicije na kojoj se nalazi element koji ima vrednost  $m$  ili  $-1$  ukoliko element nije u nizu.
- (d) Napisati funkciju koja proverava da li elementi niza čine palindrom.
- (e) Napisati funkciju koja proverava da li su elementi niza uređeni neopadajuće.

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
8 11 -2 14 -2 11 8
Ucitani niz: 8 11 -2 14 -2 11 8
Unesite jedan ceo broj:
11
Niz sadrzi element cija je vrednost 11.
Niz sadrzi element cija je vrednost 11.
Indeks njegovog prvog pojavljivanja u nizu je 1.
Niz sadrzi element cija je vrednost 11.
Indeks njegovog poslednjeg pojavljivanja u nizu je 5.
Elementi niza cine palindrom.
Niz nije sortiran neopadajuće.
```

#### 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 sve vrednosti niza uvećava za zadatu vrednost  $m$ .
- (b) Napisati funkciju koja obrće elemente niza.
- (c) Napisati funkciju koja rotira niz ciklično za jedno mesto u levo.
- (d) 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
Unesite jedan ceo broj:
2
Elementi niza nakon uvecanja za 2:
9 -1 13 785 28 -17
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]);
32     }
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]);
40         }
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

```

1  #include <stdio.h>
3  #define BROJ_CIFARA 10
5  int main()
{
7     char c;
    int cifrex[BROJ_CIFARA], cifrey[BROJ_CIFARA];
9     int x, y, i, indikator;

11     /* Ucitavaju se brojevi x i y */
    printf("Unesite dva broja: ");
13     scanf("%d%d", &x, &y);

15     /* Za slucaj da su unete vrednosti negativne, posmatra se njihova
        apsolutna vrednost.
        Ovo je opravdano iz razloga sto se brojevi x i -x zapisuju istim
        ciframa. */
17     x=abs(x);
    y=abs(y);
19

    /* Niz cifrex predstavlja brojace za cifre broja x.
       Niz cifrey predstavlja brojace za cifre broja y.
       Na pocetku se ovi nizovi inicijalizuju nulama. */
23     for(i=0; i<BROJ_CIFARA; i++)
    {
25         cifrex[i] = 0;
        cifrey[i] = 0;
27     }

29     /* Analiziraju se cifre broja x */
    while(x)
31     {
        c = x%10;
33         cifrex[c]++;
        x /= 10;
35     }

37     /* Analiziraju se cifre broja y */
    while(y)
39     {
        c = y%10;
41         cifrey[c]++;

```

```

43     y /= 10;
44 }
45
46 /* Promenljiva indikator služi za praćenje da li su oba broja
47    sastavljena od istih cifara. */
48 indikator = 1;
49
50 for(i=0; i<BROJ_CIFARA; i++){
51     /* Ako se broj pojavljivanja cifre i u zapisu broja x razlikuje
52        od broja pojavljivanja cifre i u
53        zapisu broja y, brojevi se ne zapisuju istim ciframa. Zato se
54        vrednost indikatora može postaviti na
55        0 i prekinuti dalje upoređivanje broja pojavljivanja. */
56     if(cifrey[i] != cifrex[i])
57     {
58         indikator = 0;
59         break;
60     }
61 }
62
63 /* Ako je vrednost promenljive indikator ostala 1, to znači da u
64    petlji nije pronađena cifra
65    koja se ne pojavljuje isti broj puta u zapisima brojeva x i y.
66    Zato se može zaključiti da se brojevi
67    zapisuju istim ciframa. */
68 if(indikator)
69     printf("Brojevi se zapisuju istim ciframa!\n");
70 else
71     printf("Brojevi se ne zapisuju istim ciframa!\n");
72
73 return 0;
74 }

```

### Rešenje 1.1.5

```

1 #include <stdio.h>
2
3 int main()
4 {
5     /* Niz u kojem će se čuvati informacije o broju pojavljivanja
6        cifara */
7     int cifre[10];
8
9     /* Niz u kojem će se čuvati informacije o broju pojavljivanja malih
10        slova */
11     int mala_slova[26];
12
13     /* Niz u kojem će se čuvati informacije o broju pojavljivanja
14        velikih slova */
15     int velika_slova[26];

```

```

14     int c, i;

16     /* Brojaci karaktera se na pocetku inicijalizuju nulama */
17     for(i=0;i<10;i++){
18         cifre[i]=0;
19     }

20     for(i=0;i<26;i++)
21     {
22         mala_slova[i]=0;
23         velika_slova[i]=0;
24     }

26     /* Ucitavaju se karakteri sve do kraja ulaza */
27     while((c = getchar()) != EOF)
28     {
29         /* Ako je procitani karakter veliko slovo ... */
30         if (c>='A' && c<='Z'){
31             velika_slova[c-'A']++;
32         }
33         else{
34             /* Ako je procitani karakter malo slovo ... */
35             if (c>='a' && c<='z'){
36                 mala_slova[c-'a']++;
37             }
38             else{
39                 /* Ako je procitani karakter cifra ... */
40                 if(c >='0' && c <= '9'){
41                     cifre[c-'0']++;
42                 }
43             }
44         }
45     }

46 }

48 /* Ispisuju se trazene informacije */
49 for(i = 0; i < 10; i++){
50     if (cifre[i]!=0)
51         printf("Karakter %c se pojavljuje %d puta\n", '0' + i, cifre[i]);
52 }

54 for(i = 0; i < 26; i++){
55     if (mala_slova[i]!=0)
56         printf("Karakter %c se pojavljuje %d puta\n", 'a' + i,
57             mala_slova[i]);
58 }

59 for(i = 0; i < 26; i++){
60     if (velika_slova[i]!=0)
61         printf("Karakter %c se pojavljuje %d puta\n", 'A' + i,
62             velika_slova[i]);

```

```
62     }
64
66     return 0;
}
```

## Rešenje 1.1.6

## Rešenje 1.1.7

```
2  #include <stdio.h>
4
6  #define MAX 100
8
10 int main()
12 {
14     int a[MAX];
16     int b[MAX];
18
20     /* Rezultujući niz ima najviše 2*MAX elemenata */
22     int c[2*MAX];
24
26     int n;
28     int i,j;
30
32     /* Učitava se dimenzija nizova i proverava njena ispravnost */
34     printf("Unesite dimenziju nizova:\n");
36     scanf("%d", &n);
38     if (n<1 || n>MAX)
40     {
42         printf("Nedozvoljena vrednost!\n");
44         return -1;
46     }
48
50     /* Učitavaju se elementi prvog niza */
52     printf("Unesite elemente niza a:\n");
54     for(i=0;i<n;i++)
56     {
58         scanf("%d", &a[i]);
60     }
62
64     /* Učitavaju se elementi drugog niza */
66     printf("Unesite elemente niza b:\n");
68     for(i=0;i<n;i++)
70     {
72         scanf("%d", &b[i]);
74     }
76
78     /*
```

```

42     Formira se treci niz.
    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");
58 for(i=0;i<2*n;i++)
    printf("%d ",c[i]);
60 printf("\n");
62 return 0;
64 }

```

### Rešenje 1.1.8

```

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
/* Ispisuju se elementi niza c */
41
for(i=0;i<2*n;i++)
    printf("%d ", c[i]);
43
printf("\n");
45
return 0;
}

```

### Rešenje 1.1.9

### Rešenje 1.1.10

```

#include <stdio.h>
2
#define MAX 100
4
int main()
6
{
8
    int a[MAX];
    int n;
10
    int i,j;
    char poslednja_cifra;
12
    int novo_n;

14
    /* Ucitava se dimenzija niza i proverava njena ispravnost */
    printf("Unesite dimenziju niza:\n");
16
    scanf("%d", &n);
    if (n<1 || n>MAX)

```

```

18 {
19     printf("Nedozvoljena vrednost!\n");
20     return -1;
21 }
22
23 /* Ucitavaju se elementi niza a */
24 printf("Unesite elemente niza a:\n");
25 for(i=0;i<n;i++)
26 {
27     scanf("%d", &a[i]);
28 }
29
30 /* Obilaze se svi elementi niza a */
31 for(i=0, j=0; i<n; i++)
32 {
33     /* Izdvaja se poslednja cifra tekućeg elementa */
34     poslednja_cifra = a[i]%10;
35
36     /* Ako je poslednja cifra 0 ili je element deljiv svojom
37     poslednjom cifrom,
38     zadržavamo ga i smestamo na poziciju j */
39     if (poslednja_cifra==0 || a[i]%poslednja_cifra==0)
40     {
41         a[j]=a[i];
42         j++;
43     }
44 }
45
46 /* Dimenzija novog niza odgovara posledjoj vrednosti broja j */
47 novo_n=j;
48
49 /* Ispisuje se rezultujući niz */
50 printf("Niz a nakon izmena:\n");
51 for(i=0; i<novo_n;i++)
52     printf("%d ", a[i]);
53 printf("\n");
54
55 return 0;
56 }

```

### Rešenje 1.1.11

### Rešenje 1.1.12

```

#include <stdio.h>

2
#define MAX 100

4

```

```

6  /* a) Napisati funkciju koja ucitava elemente niza. */
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 stampaj(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;
28     int s=0;
    for(i=0;i<n;i++)
30     s+=a[i];
    return s;
32 }

34 /* d) Napisati funkciju koja racuna prosechnu vrednost elemenata niza.
    */
36 float prosek(int a[], int n)
{
38     int i;
    int s = suma(a,n);
40     return (float) s/n;
}

42

44 /* e) Napisati funkciju koja izracunava minimum elemenata niza.*/
int minimum (int a[],int n)
46 {
    int m;
48     int i;

50     /* Minimum inicijalizujemo prvim elementom niza (a[0]), a zatim
        prolazimo kroz ostatak niza.
        U svakom koraku poredimo vrednost minimuma sa tekucim elementom
        niza. */
52     m = a[0];
    for(i=1;i<n;i++)

```

```

54         if (a[i] < m)
55             m = a[i];
56
57         /* Vraca se izracunata vrednost minimuma */
58         return m;
59     }
60
61
62     /* f) Napisati funkciju koja izracunava poziciju maksimalnog elementa
        u nizu. */
63     int pozicija_maksimuma (int a[],int n)
64     {
65         int m;
66         int m_pozicija;
67         int i;
68
69         /* Maksimum inicijalizujemo prvim elementom niza (a[0]), a zatim
            prolazimo kroz ostatak niza.
            U svakom koraku poredimo vrednost maksimuma sa tekucim
            elementom niza. */
70
71         m = a[0];
72         m_pozicija=0;
73         for(i=1;i<n;i++)
74             if (a[i] > m)
75             {
76                 m = a[i];
77                 m_pozicija=i;
78             }
79
80         /* Vraca se izracunata pozicija */
81         return m_pozicija;
82     }
83
84
85
86     int main()
87     {
88         int a[MAX];
89         int n;
90
91         /* Ucitava se dimenzija niza i proverava njena ispravnost */
92         printf("Unesite dimenziju niza:");
93         scanf("%d",&n);
94         if (n<1 || n>MAX)
95         {
96             printf("Nedozvoljena vrednost!\n");
97             return -1;
98         }
99
100
101         /* Testira se funkcija kojom se ucitavaju elementi niza */

```

```

104     ucitaj(a,n);

106     /* Testira se funkcija kojom se ispisuju elementi niza */
    printf("Ucitani niz: ");
    stampaaj(a,n);

108     /* Testira se funkcija kojom se izracunava suma elemenata niza */
110     printf("Suma elemenata niza: %d\n", suma(a,n));

112     /* Testira se funkcija kojom se racuna prosek elemenata niza */
    printf("Prosecna vrednost elemenata niza: %.2f\n", prosek(a,n));

114     /* Testira se funkcija kojom se izracunava minimum niza */
116     printf("Minimumalni element niza: %d\n", minimum(a,n));

118     /* Testira se funkcija kojom se izracunava pozicija maksimalnog
        elementa */
    printf("Indeks maksimalnog elementa niza: %d\n",
        pozicija_maksimuma(a,n));

120     return 0;
122 }

```

### 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 stampaaj(int a[], int n)
17 {
18     int i;
19     for(i=0;i<n;i++)
20     {
21         printf("%d ",a[i]);
22     }
23     printf("\n");
24 }
25
26 /* a) Funkcija koja proverava da li niz sadrzi zadatu vrednost m */
27 int sadrzi(int a[], int n, int m)

```

```

28     int i;

30     /* Poredi se element po element niza a sa zadatim brojem m */

32     for(i=0;i<n;i++){
33         /* Ukoliko je tekuci element niza jednak trazenom broju */
34         if (a[i]==m){
35             /* Funkcija vraca vrednost 1 */
36             return 1;
37         }
38     }

40     /* Ako se stigne do kraja niza i ne naidje na vrednost koja je
       jednaka broju m,
       to znaci da se broj ne nalazi u nizu i da funkcija treba da
       vrati 0. */
42     return 0;
43 }

44 /* b) Funkcija koja vraca vrednost prve pozicije na kojoj se nalazi
       element koji ima vrednost m,
       ili -1 ukoliko element nije u nizu */
46 int prvo_pojavljivanje(int a[], int n, int m)
47 {
48     int i;

50     /* Poredi se element po element niza a sa zadatim brojem m */

52     for(i=0;i<n;i++){
53         /* Ukoliko je tekuci element niza jednak trazenom broju */
54         if (a[i]==m){
55             /* Vraca se njegov indeks */
56             return i;
57         }
58     }

60     /* Ako se stigne do kraja niza i ne naidje na vrednost koja je
       jednaka broju m,
       to znaci da se broj ne nalazi u nizu i da funkcija treba da
       vrati -1. */
62     return -1;
63 }

64 /* c) Funkcija koja vraca vrednost poslednje pozicije na kojoj se
       nalazi element koji ima vrednost m,
       ili -1 ukoliko element nije u nizu */
66 int poslednje_pojavljivanje(int a[], int n, int m)
67 {
68     int i;

70     /* Polazi se od kraja niza i poredi se element po element sa
71

```

```

    zadan broj m*/
74     for(i=n-1;i>=0;i--){
        /* Ukoliko je tekuci element niza jednak trazenom broju */
76         if (a[i]==m){
            /* Vraca se njegov indeks */
78             return i;
        }
80     }

82     /* Ako se stigne do pocetka niza i ne naide na vrednost koja je
        jednaka broju m,
        to znaci da se broj ne nalazi u nizu i da funkcija treba da
        vrati -1. */
84     return -1;
}

86 /* d) Funkcija koja proverava da li elementi niza cine palindrom */
88 int palindrom(int a[], int n)
89 {
90     int i,j;
91
92     /*
93     Uporedjuje se element na poziciji 0 sa elementom na poziciji n-1
94     Uporedjuje se element na poziciji 1 sa elementom na poziciji n-2
95     Uporedjuje se element na poziciji 2 sa elementom na poziciji n-3
96     .
97     .
98     i tako redom dok je pozicija prvog elementa manja od pozicije
99     drugog elementa
100    */

102    for(i=0,j=n-1;i<j;i++,j--){
        /* Ako element na poziciji i nije jednak odgovarajucem elementu
           na poziciji j */
104        if(a[i]!=a[j]){
            /* Moze se odmah zakljuciti da niz nije palindrom */
106            return 0;
        }
108    }

110    /* Ako su svi parovi elemenata jednaki, niz je palindrom i
        funkcija vraca vrednost 1 */
    return 1;
112 }

114 /* e) Funkcija koja proverava da li su elementi niza uredjeni
    neopadajuce */
int neopadajuci(int a[], int n)
116 {
    int i;

```

```

118      /* Ako je niz uredjen neopadajuće vase nejednakosti:
120         a[0]<=a[1], a[1]<=a[2], ..., a[i]<=a[i+1], ..., a[n-2]<=a[n-1]
         Zato je dovoljno proveriti da li za parove susednih elemenata
         vazi ovi svojstvo.
122     */
123     for(i=0; i<n-1; i++)
124         if (a[i]>a[i+1])
125             return 0;
126
127     return 1;
128 }
129
130 /* f) Funkcija koja izracunava najduzu uzastopnu seriju jednakih
131    elemenata u nizu. Na primer, za uneti niz 1 2 3 4 4 5 6 7 8 9 9
132    funkcija
133    treba da vrati 3. */
134 int najduza_serija_jednakih(int a[], int n)
135 {
136     int i;
137     int j;
138     int duzina;
139     int max_duzina=0;
140
141
142     for(i=0,j=0;i<n-1;i++)
143     {
144         if(a[i]==a[i+1])
145         {
146             j++;
147
148             /*
149                ako se niz završava konstantnim intervalom (nalazimo se u
150                poslednjoj
151                iteraciji petlje i tada je i==n-2), ispitujemo da li je
152                taj konstantni
153                interval maksimalne duzine
154            */
155            if(i==n-2)
156            {
157                j++;
158                if(j>max_duzina)
159                    max_duzina=j;
160            }
161        }
162        else
163        {
164            /*
165                izašli smo iz konstantnog intervala
166
167                ukoliko smo imali bar dva elementa u konstantnom

```



```

166     intervalu,
167     vrednost promenljive j ce biti 1, a duzina tog intervala
168     je 2;
169     zbog toga je neophodno takve (pozitivne) j uvecati za 1;
170
171     sa druge strane, ako su a[i] i a[i+1] razliciti,
172     duzina tog intervala je 0
173     */
174
175     if (j>0)
176         j++;
177
178     /* azuriramo maksimalnu duzinu uspona */
179     if(j>max_duzina)
180         max_duzina=j;
181     /*
182     duzina uspona se postavlja na nulu
183     kako bi mogli da je iskoristimo
184     za naredni uspon
185     */
186     j=0;
187
188 }
189
190 }
191
192 return max_duzina;
193 }
194
195 int main()
196 {
197     int a[MAX];
198     int n;
199     int m;
200     int i;
201
202     /* Ucitava se dimenzija niza i proverava se njena ispravnost */
203     printf("Unesite dimenziju niza:");
204     scanf("%d",&n);
205     if (n<1 || n>MAX)
206     {
207         printf("Nedozvoljena vrednost!\n");
208         return -1;
209     }
210
211     /* Ucitavaju se i ispisuju elementi niza */
212     ucitaj(a,n);
213     printf("Ucitani niz:");
214     stampaj(a,n);

```

```

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

220      /* I proverava se rad napisanih funkcija */
if(sadrzi(a,n,m))
222          printf("Niz sadrzi element cija je vrednost %d\n", m);
else
224          printf("Niz ne sadrzi element cija je vrednost %d\n", m);

226      i = prvo_pojavljivanje(a,n,m);
if(i!=-1)
228          printf("Niz sadrzi element cija je vrednost %d. Indeks njegovog
prvog pojavljivanja u nizu je %d\n", m,i);
else
230          printf("Niz ne sadrzi element cija je vrednost %d\n", m);

232
234      i = poslednje_pojavljivanje(a,n,m);
if(i!=-1)
236          printf("Niz sadrzi element cija je vrednost %d. Indeks njegovog
poslednjeg pojavljivanja u nizu je %d\n", m,i);
else
238          printf("Niz ne sadrzi element cija je vrednost %d\n", m);

240      if(palindrom(a,n))
printf("Elementi niza cine palindrom\n");
else
242          printf("Elementi niza ne cine palindrom\n");

244      if(neopadajuci(a,n))
printf("Niz je sortiraneopadajuce\n");
else
246          printf("Niz nije sortiraneopadajuce\n");

248      printf("Duzina najduzeg konstantnog intervala: %d\n",
najduza_konstanta(a,n));

250
252      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;
9      for(i=0;i<n;i++)
10     {
11         scanf("%d",&a[i]);
12     }
13 }
14
15 /* Funkcija kojom se ispisuju elementi niza a dimenzije n */
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 sve vrednosti niza uvecava za zadatu vrednost m
25    */
26 void uvecaj(int a[], int n, int m)
27 {
28     int i;
29     for(i=0;i<n;i++)
30         a[i]+=m;
31 }
32
33 /* b) Funkcija koja obrce elemente niza */
34 void obrni(int a[], int n)
35 {
36     int t;
37     int i,j;
38
39     /*
40      Za niz a[0], a[1], ..., a[n-2], a[n-1] obrnuti niz je a[n-1], a[
41      n-2], ..., a[1], a[0]
42      Zato je potrebno razmeniti vrednosti elemenata a[0] i a[n-1], a
43      [1] i a[n-2], itd. i zaustaviti se
44      kada je vrednost indeksa prvog elementa veca od vrednosti drugog
45      elementa.
46      */
47     for(i=0,j=n-1;i<j;i++, j--)
48     {
49         t = a[i];
50         a[i] = a[j];
51         a[j] = t;
52     }
53 }
54
55 /* c) Funkcija koja rotira niz ciklicno za jedno mesto u levo */

```

```

56 void rotiraj1(int a[], int n)
57 {
58     int i;
59     int tmp;
60
61     /* Izdvaja se prvi element niza */
62     tmp=a[0];
63
64     /* Pomeraju se preostali elementi niza */
65     for(i=0;i<n-1;i++){
66         a[i]=a[i+1];
67     }
68
69     /* Poslednjem elementu se dodeljuje sacuvana vrednost prvog
70     elementa */
71     a[n-1] = tmp;
72 }
73
74 /* d) Funkcija koja rotira niz ciklicno za k mesta u levo */
75 void rotirajk(int a[], int n, int k)
76 {
77     int i;
78
79     /* Odredjuje se vrednost broja k koja je u opsegu od 0 do n-1 kako
80     bi se izbegla suvisna pomeranja */
81     k=k%n;
82
83     /* Niz se rotira za jednu poziciju ulevo k puta */
84     for(i=0;i<k;i++)
85         rotiraj1(a,n);
86 }
87
88 int main()
89 {
90     int a[MAX];
91     int n;
92     int i;
93     int k;
94     int m;
95
96     /* Ucitava se dimenzija niza i proverava se njena ispravnost */
97     printf("Unesite dimenziju niza:");
98     scanf("%d",&n);
99     if (n<1 || n>MAX)
100     {
101         printf("Nedozvoljena vrednost!\n");
102         return -1;
103     }
104
105     /* Ucitavaju se elementi niza */
106     ucitaj(a,n);

```

```

106      /* Testira se rad napisanih funkcija */
108
109      /* a) */
110      printf("Unesite jedan ceo broj:");
111      scanf("%d", &m);
112      printf("Elementi niza nakon uvecanja za %d:\n",m);
113      uvecaj(a,n,m);
114      stampaj(a,n);
115
116      /* b) */
117      printf("Elementi niza nakon obrtanja:\n");
118      obrni(a,n);
119      stampaj(a,n);
120
121      /* c) */
122      printf("Elementi niza nakon rotiranja za 1 mesto ulevo:\n");
123      rotiraj1(a,n);
124      stampaj(a,n);
125
126      /* d) */
127      printf("Unesite jedan pozitivan ceo broj:");
128      scanf("%d",&k);
129      if (k<=0)
130      {
131          printf("Nekorektan unos\n");
132          return -1;
133      }
134      rotirajk(a,n,k);
135      printf("Elementi niza nakon rotiranja za %d mesto ulevo:\n",k);
136      stampaj(a,n);
137
138      return 0;
139  }
140

```

## Rešenje 1.1.15

```

1  #include <stdio.h>
2
3  #define MAX 100
4
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)

```

```

15     {
16         printf("Nedozvoljena vrednost!\n");
17         return -1;
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     return 0;
38 }

```

## 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
25  /* Ucitava se broj k i proverava se njegova ispravnost */
printf("Unesite broj k: ");
27  scanf("%d", &k);
if(k == 0)
29  {
    printf("Greska: Pogresan unos!\n");
31  return -1;
}
33
/*
35  Promenljiva koja cuva informaciju o tome da li je u nizu
    postojao element koji je deljiv brojem k.
    Inicijalna vrednost je 0.
37  */
39  indikator = 0;
41
/*
    Ukoliko je element niza deljiv brojem k, indikator se postavlja
    na 1
43  i ispisuje se indeks tog elementa.
*/
45
for(i=0;i<n;i++){
47  if(brojevi[i]%k == 0)
    {
49      indikator = 1;
      printf("%d ",i);
51  }
}
53
/* Ukoliko je indikator jednak nuli to znaci da ne postoji element
    u nizu koji je deljiv brojem k. */
55
if(indikator == 0){
57  printf("U nizu nema elemenata koji su deljivi brojem %d!\n",k);
}
59
61  return 0;
}

```

## Rešenje 1.1.17

```

1  #include <stdio.h>
3  #define MAX 100
5  int main()

```

```

{
7   int brojevi[MAX];
   int n, i, poz_max, poz_min, max, min, tmp;

9   /* Ucitava se dimenzija niza i proverava se njena ispravnost */
11  printf("Unesite dimenziju niza: ");
   scanf("%d", &n);
13  if(n<1 || n>MAX)
   {
15      printf("Greska: Nedozvoljena vrednost!\n");
      return -1;
17  }

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

23  /*
25     Maksimalnim tj. minimalnim elementom niza proglašava se nulti
       element niza.
       Pozicije maksimalnog tj. minimalnog elementa se postavljaju na 0.
27  */
   max = brojevi[0];
29   min = brojevi[0];
   poz_max = 0;
31   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       }

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

49 /* Zamenjuju se elementi na pozicijama poz_min i poz_max */
   tmp = max;
51   brojevi[poz_max] = min;
   brojevi[poz_min] = tmp;
53
   /* Ispisuje se rezultujući niz */
55   for(i=0;i<n;i++)

```



```
        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
12
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 == '*')
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 }
```

```
    }
39     printf("\n");
41
42     return 0;
43 }
```

## 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 */
16         if(niz[i] == x){
17             /* Uvecava se broj pojavljivanja */
18             brojac++;
19         }
20     }
21
22     /* Vraca se izracunata vrednost */
23     return brojac;
24 }
25
26 int main()
27 {
28     /* Niz elemenata koje zadaje korisnik */
29     int a[MAX];
30
31     /* Niz elemenata koji se pojavljuju tri puta */
32     int b[MAX];
33
34     int i, j, n, n_b;
35
36     /* Ucitava se broj elemenata korisnickog niza i proverava se
37     njegova ispravnost */
38     printf("Unesite broj n: ");
39     scanf("%d", &n);
40     if(n<1 || n>MAX)
41     {
42         printf("Greska: Nedozvoljena vrednost!\n");
43     }
44 }
```

```

43     return -1;
44 }
45
46 /* Ucitavaju se elementi korisnickog niza */
47 printf("Unesite elemente niza a: ");
48 for(i=0;i<n;i++)
49     scanf("%d", &a[i]);
50
51 /* j - broj elementa rezultujucega niza b */
52 j = 0;
53
54 /* Obilazi se element po element niza a */
55 for(i=0;i<n;i++)
56 {
57     /* Ukoliko se tekuci element pojavljuje tacno tri puta u nizu a i
58        nije upisan u niz b
59        koji trenutno ima j elemenata, dodaje se u niz b na poziciju j i
60        uvecava se broj elemenata niza b */
61     if(broj_pojavljivanja(a, n, a[i])==3 && broj_pojavljivanja(b, j,
62        a[i])==0)
63     {
64         b[j] = a[i];
65         j++;
66     }
67 }
68
69 /* Ispisuje se rezultujući niz b - broj elemenata u nizu b je j*/
70 n_b = j;
71 for(i=0;i<n_b;i++)
72     printf("%d ", b[i]);
73 printf("\n");
74
75 return 0;
76 }

```

### Rešenje 1.1.21

### Rešenje 1.1.22

```

1 #include <stdio.h>
2
3 #define MAX 100
4
5 /*
6  Funkcija koja vraca 1 ukoliko broj x postoji u nizu, 0 inace.
7  */
8
9 int postoji(int niz[], int n, int x)
10 {
11     int i;

```

```

12     for(i=0;i<n;i++)
13         if(niz[i] == x)
14             return 1;
15
16     return 0;
17 }
18
19
20 int main()
21 {
22     int a[MAX], b[MAX], unija[2*MAX], presek[MAX], razlika[MAX];
23     int i, j, n_a, n_b, n_u, n_p, n_r, indikator;
24
25     printf("Unesite broj elemenata niza a: ");
26     scanf("%d", &n_a);
27
28     if(n_a<1 || n_a>100)
29     {
30         printf("Greska: pogresan unos!\n");
31         return -1;
32     }
33
34     printf("Unesite elemente niza a: ");
35     for(i=0;i<n_a;i++)
36         scanf("%d", &a[i]);
37
38     printf("Unesite broj elemenata niza b: ");
39     scanf("%d", &n_b);
40
41     if(n_b<1 || n_b>100)
42     {
43         printf("Greska: pogresan unos!\n");
44         return -1;
45     }
46
47     printf("Unesite elemente niza b: ");
48     for(i=0;i<n_b;i++)
49         scanf("%d", &b[i]);
50
51     /*
52      Brojaci elemenata u nizovima unija, presek i razlika.
53     */
54     n_u = 0;
55     n_p = 0;
56     n_r = 0;
57
58     for(i=0;i<n_a;i++)
59     {
60         /*
61          Ukoliko se element a[i] ne nalazi u uniji, dodajemo ga u uniju
62          i povecamo brojac elemenata u nizu unija.
63         */

```

```

64     if(postoji(unija,n_u,a[i]) == 0)
65     {
66         unija[n_u] = a[i];
67         n_u++;
68     }
69
70     /*
71      Ukoliko se element a[i] postoji u nizu b i ne postoji u nizu
72      presek, dodajemo ga u presek i povecavamo brojac elemenata u nizu
73      presek.
74      */
75     if(postoji(b, n_b, a[i])==1 && postoji(presek, n_p, a[i])==0)
76     {
77         presek[n_p] = a[i];
78         n_p++;
79     }
80
81     /*
82      Ukoliko element a[i] ne postoji u nizu b i ne postoji u nizu
83      razlika, dodajemo ga u razliku i povecavamo brojac elemenata u
84      nizu razlika.
85      */
86     if(postoji(b, n_b, a[i])==0 && postoji(razlika, n_r, a[i])==0)
87     {
88         razlika[n_r] = a[i];
89         n_r++;
90     }
91 }
92
93 /*
94  Elemente niza b koji ne postoje u uniji dodajemo u uniju.
95  */
96 for(i=0;i<n_b;i++)
97     if(postoji(unija, n_u, b[i]))
98     {
99         unija[n_u] = b[i];
100         n_u++;
101     }
102
103 printf("Unija: ");
104 for(i=0;i<n_u;i++)
105     printf("%d ", unija[i]);
106
107 printf("\nPresek: ");
108 for(i=0;i<n_p;i++)
109     printf("%d ", presek[i]);
110
111 printf("\nRazlika: ");
112 for(i=0;i<n_r;i++)
113     printf("%d ", razlika[i]);
114
115 return 0;

```

```
}
```

### Rešenje 1.1.23

```
#include <stdio.h>

2
#define MAX 100
4
int main()
6 {
    int a[MAX];
8     int i, j, n_a;

10     /* Ucitava se broj elemenata niza i proverava se njegova ispravnost
        */
    printf("Unesite broj elemenata niza: ");
12     scanf("%d", &n_a);
    if(n_a<1 || n_a>100)
14     {
        printf("Greska: Nedozvoljena vrednost!\n");
16         return -1;
    }

18     /* Ucitavaju se elementi niza */
    printf("Unesite elemente niza: ");
20     for(i=0; i<n_a; i++)
        scanf("%d", &a[i]);
22

24
    /*
26     1. nacin

28     int b[MAX], n_b;

30     // Brojac j predstavlja poziciju u nizu b na koju treba smestiti
        element niza a. Njegova pocetna vrednost je 0.

32     for(i=0, j=0; i<n_a; i++){
        // Ako je tekuci element niza a paran
34         if(a[i]%2 == 0)
        {
36             // Smesta se na poziciju j u nizu b
                b[j] = a[i];
38             // Vrednost brojaca j se priprema za narednu iteraciju
                j++;
40         }

42         // Ako je element niza a neparan, sa njim nista ne treba raditi
    }

44     // Broj elemenata novodobijenog niza b je j
```

```

46     n_b = j;

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

52     */

54     /*
    2. nacin

56     J predstavlja brojac prve slobodne pozicije na koju se moze
        upisati element niza koji treba da ostane u nizu.
58     Kada se naidje na element koji je paran, on se kopira na mesto a[j]
        i poveca se vrednost brojac j.
        Ukoliko se naidje na element koji je neparan, njega treba
        preskociti.
60     */
    for(i=0, j=0;i<n_a;i++)
62     {
        /* Ako je tekuci element niza a paran */
64         if(a[i]%2 == 0)
        {
66             /* Premesta se na poziciju j */
            a[j] = a[i];

68             /* Vrednost brojac j se priprema za narednu iteraciju */
70             j++;
        }

72         /* Ako je tekuci element niza a neparan, sa njim nista ne treba
            raditi */
74     }

76     /* U nizu a se sada na pozicijama od 0,...,j-1 nalaze elementi koji
        su parni, te je njegova nova dimenzija j. */
    n_a=j;

78     /* Ispisuju se elementi modifikovanog niza a */
    for(i=0;i<n_a;i++){
80         printf("%d ", a[i]);
82     }
    printf("\n");

84     return 0;
86 }

```

## Rešenje 1.1.24

```

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

```

```

4 #define MAX 100

6 /*
8  Funkcija koja proverava da li je zadati broj prost broj.
9  Povratna vrednost funkcije je 1 ukoliko broj jeste prost, inace je
10  0.
11 */
12 int prost(int x)
13 {
14     int i;

15     /* Posmatra se apsolutna vrednost broja kako bi se pokrio i slucaj
16        negativnih brojeva */
17     x=abs(x);

18     /* Brojevi 1, 2 i 3 su prosti */
19     if(x == 1 || x == 2 || x == 3)
20         return 1;

21     /* Ako je broj paran nije prost */
22     if(x%2 == 0)
23         return 0;

24     /* Ako broj ima delioce u skupu [3, koren_broja(x)] takodje nije
25        prost */
26     for(i=3;i<=sqrt(x);i+=2){
27         if(x%i == 0)
28             return 0;
29     }

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

34 int main()
35 {
36     int a[MAX];
37     int i, j, n_a, n_b;

38     /* Ucitava se broj elemenata niza i proverava se njegova ispravnost
39        */
40     printf("Unesite broj elemenata niza: ");
41     scanf("%d", &n_a);
42     if(n_a<1 || n_a>MAX)
43     {
44         printf("Greska: Nedozvoljena vrednost!\n");
45         return -1;
46     }

47     /* Ucitavaju se elementi niza a */
48     printf("Unesite elemente niza: ");

```



```

52     for(i=0;i<n_a;i++)
        scanf("%d", &a[i]);

54
56     /*
57     1. nacin
58     int b[MAX];

60
61     for(i=0, j=0;i<n_a;i++){
62         if(prost(a[i]) == 0)
63         {
64             b[j] = a[i];
65             j++;
66         }
67     }

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

71
72     // Ispisuju se elementi niza b
73     for(i=0;i<n_b;i++)
74         printf("%d ", b[i]);
75     printf("\n");

76
77     */
78
79     /*
80     2. nacin
81     */
82
83     for(i=0, j=0; i<n_a; i++)
84     {
85         if(prost(a[i]) == 0)
86         {
87             a[j] = a[i];
88             j++;
89         }
90     }

91     n_a = j;

92
93     /* Ispisuju se elementi modifikovanog niza a */
94     for(i=0;i<n_a;i++)
95         printf("%d ", a[i]);
96     printf("\n");

97
98     return 0;
99 }
100

```

## 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      int i;
10
11     /* Brojac elemenata koji su manji od poslednjeg */
12     int broj_manjih=0;
13
14     /* Obilazi se element po element niza */
15     for(i=0;i<n-1;i++){
16         /* Ako je tekuci element manji od poslednjeg (on se nalazi na
17            poziciji n-1) */
18         if(a[i]<a[n-1]){
19             /* Uvecava se brojac */
20             broj_manjih++;
21         }
22     }
23
24     /* Vraca se izracunata vrednost */
25     return broj_manjih;
26 }
27
28 int main()
29 {
30     int a[MAX];
31     int n;
32     int i;
33
34     /* Ucitava se broj elemenata niza i proverava se njegova ispravnost
35        */
36     printf("Unesite broj elemenata niza:");
37     scanf("%d", &n);
38     if(n<=0 || n>MAX)
39     {
40         printf("Greska: Nedozvoljena vrednost!\n");
41         return 0;
42     }
43
44     /* Ucitavaju se elementi niza */
45     printf("Unesite elemente niza:");
46     for(i=0;i<n;i++)
47         scanf("%d",&a[i]);
48
49     /* Ispisuje se rezultat poziva funkcije */
50     printf("%d\n", prebrojavanje(a,n));
```

```
48     return 0;
50 }
```

## Rešenje 1.1.26

```
1  #include <stdio.h>
2
3  #define MAX 100
4
5  /* Funkcija vraca broj parnih elemenata niza koji prethode
6     maksimalnom elementu niza */
7  int prebrojavanje(int a[], int n)
8  {
9      int i;
10
11     int maksimum;
12     int pozicija_maksimuma;
13
14     /* Brojac elemenata koji su parni i prethode maksimalnom */
15     int broj_parnih;
16
17     /* Pronalazi se maksimalni element niza i njegova pozicija */
18     maksimum = a[0];
19     pozicija_maksimuma = 0;
20
21     for(i=1; i<n-1; i++)
22         if(a[i] > maksimum)
23         {
24             maksimum = a[i];
25             pozicija_maksimuma = i;
26         }
27
28     /* Prebrojavaju se parni elementi koji prethode maksimalnom */
29     broj_parnih = 0;
30     for(i=0; i < pozicija_maksimuma; i++){
31         if(a[i]%2==0){
32             broj_parnih++;
33         }
34     }
35
36     /* Vraca se izracunata vrednost */
37     return broj_parnih;
38 }
39
40 int main()
41 {
42     int a[MAX];
43     int n;
44     int i;
```

```

/* 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
   #define MAX 100
5
/* Funkcija prebrojava cifre u datom nizu karaktera */
7 int cifre(char a[], int n)
{
9     int i;

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

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

41  /* Ucitavaju se elementi niza */
43  printf("Unesite elemente niza:");
    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 */
        for(k=i; k<=j; k++){
13         z+=a[k];
15     }

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

```

```

19 }
21 int main(){
23     int n, i, j;
24     int a[MAX];
25
26     /* Ucitava se broj elemenata niza i proverava se njegova ispravnost */
27     printf("Unesite broj elemenata niza: ");
28     scanf("%d", &n);
29     if(n <= 0 || n > MAX)
30     {
31         printf("Greska: Nedozvoljena vrednost!\n");
32         return 0;
33     }
34
35     /* Ucitavaju se elementi niza */
36     printf("Unesite elemente niza:");
37     for(i=0; i<n; i++)
38         scanf("%d", &a[i]);
39
40     /* Ucitavaju se vrednosti granica */
41     printf("Unesite vrednosti za i i j: ");
42     scanf("%d%d", &i, &j);
43
44     /* Proverava se korektnost zadatog intervala */
45     if(i < 0 || j < 0 || i > n-1 || j > n-1 || i > j){
46         printf("Greska: Nekorektne vrednosti granica!\n");
47         return 0;
48     }
49
50     /* Ispisuje se rezultat poziva funkcije */
51     printf("Zbir je: %d", zbir(a,n,i,j));
52
53     return 0;
54 }

```

### Rešenje 1.1.29

```

1 #include<stdio.h>
2
3 #define MAX 100
4
5 /* Funkcija racuna zbir prvih k pozitivnih elemenata niza */
6 float zbir_pozitivnih(float a[], int n, int k){
7
8     int i;
9
10    /* Zbir pozitivnih elemenata */
11    float zbir=0;

```

```

12      /* Obilazi se element po element niza - postupk se završava ukoliko
13         se dodje do kraja niza
14         ili ukoliko se sabere k pozitivnih elemenata */
15     for(i=0; i<n && k>0; i++){
16         /* Ako je tekuci element pozitivan broj */
17         if(a[i] >= 0){
18             /* Dodaje se zbiru */
19             zbir+=a[i];
20             /* I umanjuje se brojac pozitivnih elemenata */
21             k--;
22         }
23     }
24
25     /* Vraca se izracunata vrednost */
26     return zbir;
27 }
28
29 int main(){
30     int n, i, k;
31     float a[MAX];
32
33     /* Ucitava se broj elemenata niza i proverava se njegova ispravnost
34        */
35     printf("Unesite broj elemenata niza: ");
36     scanf("%d", &n);
37     if(n<=0 || n> MAX){
38         printf("Greska: Nedozvoljena vrednost!\n");
39         return 0;
40     }
41
42     /* Ucitavaju se elementi niza */
43     printf("Unesite elemente niza: ");
44     for(i=0; i<n; i++){
45         scanf("%f", &a[i]);
46     }
47
48     /* Ucitava se broj k i proverava se njegova ispravnost */
49     printf("Unesite vrednost za k: ");
50     scanf("%d", &k);
51     if(k<0 || k>n){
52         printf("Greska: Nedozvoljena vrednost!");
53         return 0;
54     }
55
56     /* Ispisuje se rezultat poziva funkcije */
57     printf("Zbir je: %.2f\n", zbir_pozitivnih(a,n,k));
58
59     return 0;
60 }

```

## 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 int main(){

19     int n, i, j;
21     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;

```



Rešenje [1.1.31](#)

Rešenje [1.1.32](#)

Rešenje [1.1.33](#)

Rešenje [1.1.34](#)

Rešenje [1.1.35](#)