

PROGRAMIRANJE 1

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

PROGRAMIRANJE 1
Zbirka zadataka sa rešenjima

**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 sa rešenjima

Sadržaj

0.1	Petlje	vii
0.1.1	Ispis podataka	vii
0.1.2	Obrada celih brojeva, rad sa ciframa broja	x
0.1.3	Unos i obrada veće količine podatka (unos i obrada niza brojeva?, nije sjajno zbog nizova)	xii
0.1.4	Rad sa karakterima	xvii
0.1.5	Računanje sume i proizvoda	xviii
0.1.6	Dvostruka petlja i ispisivanje slike	xxii
0.2	Rešenja	xxx

Predgovor

U okviru kursa *Programiranje 1* na Matematičkom fakultetu vežbaju se zadaci koji imaju za cilj da studente nauče ...

Autori

0.1 Petlje

0.1.1 Ispis podataka

REDOSLED: Petlje se sustinski koriste za tri stvari: map, filter i reduce, kao i za kombinaciju te tri stvari.

Map — preslikavanje, dakle ceo niz necega se preslikava na neki nacin u neki novi niz (dupliranje vrednosti svih elemenata niza, dupliranje svake cifre broja, dodavanje prefiksa svim recima...)

Filter — iz niza necega biraju se neki koji zadovoljavaju neki kriterijum (svi parni brojevi, svi koji sadrze karakter "a", svi prosti brojevi, svi savrseni brojevi...)

Reduce — ceo niz se svodi na jednu vrednost (zbir svih vrednost, proizvod svih vrednosti, nadovezane sve vrednosti...)

Kombinacija — dve tehnike od prethodne tri (npr filter-reduce: zbir svih parnih brojeva) ili od svake po malo (zbir svih dupliranih brojeva koji su savrseni)

Sustinski, studenti treba da usvoje najpre ove tri tehnike, pa onda da idu ne njihove kombinacije, i to najpre na kombinacije dve od tri, pa na kraju na zadatke koje kombinuju sve to. Ove tehnike nisu vezane za nizove, mogu se primeniti i na prirodne brojeve posmatrane kao niz brojeva ili na prirodni broj posmatran kao niz cifara...

Danijela:: ok neka bude redosled koji si predložila

Zadatak 0.1 Napisati program koji 5 puta ispisuje tekst `Mi volimo da programiramo.`

Primer 1

```
|| INTERAKCIJA SA PROGRAMOM:  
|| Mi volimo da programiramo.  
|| Mi volimo da programiramo.  
|| Mi volimo da programiramo.  
|| Mi volimo da programiramo.  
|| Mi volimo da programiramo.
```

[Rešenje 0.1]

Zadatak 0.2 Napisati program koji učitava ceo broj n i ispisuje n puta tekst Mi volimo da programiramo.

Primer 1

```
|| INTERAKCIJA SA PROGRAMOM:  
|| Unesite ceo broj: 6  
|| Mi volimo da programiramo.  
|| Mi volimo da programiramo.  
|| Mi volimo da programiramo.  
|| Mi volimo da programiramo.  
|| Mi volimo da programiramo.  
|| Mi volimo da programiramo.
```

Primer 2

```
|| INTERAKCIJA SA PROGRAMOM:  
|| Unesite ceo pozitivan broj 0
```

[Rešenje 0.2]

Zadatak 0.3 Napisati program koji učitava pozitivan ceo broj n a potom ispisuje sve cele brojeve od 0 do n .

Primer 1

```
|| INTERAKCIJA SA PROGRAMOM:  
|| Unesite ceo pozitivan broj: 4  
|| 0 1 2 3 4
```

Primer 2

```
|| INTERAKCIJA SA PROGRAMOM:  
|| Unesite ceo pozitivan broj: -10  
|| Neispravan unos. Promenljiva mora biti  
|| pozitivna!
```

[Rešenje 0.3]

Zadatak 0.4 Napisati program koji učitava dva cela broja n i m ispisuje sve cele brojeve iz intervala $[n, m]$.

- (a) Koristiti `while` petlju.
- (b) Koristiti `for` petlju.
- (c) Koristiti `fo-while` petlju.

Primer 1

```
|| INTERAKCIJA SA PROGRAMOM:  
|| Unesite dva cela broja: -2 4  
|| -2 -1 0 1 2 3 4
```

Primer 2

```
|| INTERAKCIJA SA PROGRAMOM:  
|| Unesite dva cela broja: 10 6  
|| Neispravan unos. Nisu dobro zadate granice  
|| intervala!
```

[Rešenje 0.4]

Zadatak 0.5 Uskladiti formulaciju zadatka sa odgovarajućom formulacijom kod nizova. Fibonačijev niz počinje ciframa 1 i 1, a svaki član se dobija zbirom prethodna dva. Napisati program koji učitava ceo neoznačen broj n i određuje i na standardni izlaz ispisuje n -ti član Fibonačijevog niza.

Primer 1

```
|| INTERAKCIJA SA PROGRAMOM:  
|| Unesite ceo broj: 10  
|| Trazeni broj je: 55
```

Primer 2

```
|| INTERAKCIJA SA PROGRAMOM:  
|| Unesite ceo broj: -100  
|| Neispravan unos. Pozicija u Fibonacijevom  
|| nizu mora biti pozitivan broj koji nije 0!
```

[Rešenje 0.5]

* **Zadatak 0.6** Niz prirodnih brojeva formira se prema sledećem pravilu:

$$a_{n+1} = \begin{cases} \frac{a_n}{2} & \text{ako je } a_n \text{ parno} \\ \frac{3 \cdot a_n + 1}{2} & \text{ako je } a_n \text{ neparno} \end{cases}$$

Napisati program koji za uneti početni član niza a_0 (ceo pozitivan broj) štampa niz brojeva sve do onog člana niza koji je jednak 1.

Primer 1

```
|| INTERAKCIJA SA PROGRAMOM:  
|| Unesite ceo broj: 56  
|| 56 28 14 7 11 17 26 13 20 10  
|| 5 8 4 2 1
```

Primer 2

```
|| INTERAKCIJA SA PROGRAMOM:  
|| Unesite ceo broj: -48  
|| Nekorektan unos. Broj mora biti pozitivan.
```

[Rešenje 0.6]

* **Zadatak 0.7** Papir A_0 ima površinu $1m^2$ i odnos stranica $1 : \sqrt{2}$. Papir A_1 dobija se podelom papira A_0 po dužoj ivici. Papir A_2 dobija se podelom A_1 papira po dužoj ivici itd. Napisati program koji za uneti neoznačen broj k ispisuje dimenzije papira A_k u milimetrima.

Primer 1

```
|| INTERAKCIJA SA PROGRAMOM:  
|| Unesite broj n: 4  
|| 297 210
```

Primer 2

```
|| INTERAKCIJA SA PROGRAMOM:  
|| Unesite broj n: 3  
|| 297 420
```

Primer 3

```
|| INTERAKCIJA SA PROGRAMOM:  
|| Unesite broj n: 7  
|| 74 105
```

Primer 4

```
|| INTERAKCIJA SA PROGRAMOM:  
|| Unesite broj n: 9  
|| 37 52
```

[Rešenje 0.62]

0.1.2 Obrada celih brojeva, rad sa ciframa broja

Zadatak 0.8 Napisati program koji učitava ceo broj i ispisuje njegove cifre u obrnutom poretaku.

[Rešenje 0.8]

Zadatak 0.9 Pravi delioci celog broja su svi delioci sem jedinice i samog tog broja. Napisati program koji učitava ceo pozitivan broj n i ispisuje sve prave delioce unetog broja. U slučaju greške pri unosu podataka ispisati odgovarajuću poruku.

[Rešenje 0.9]

Zadatak 0.10 Sa standardnog ulaza unosi se ceo neoznačen broj. Napisati program koji proverava i ispisuje da li se cifra 5 nalazi u njegovom zapisu.

Primer 1

```
|| INTERAKCIJA SA PROGRAMOM:  
|| Unesite broj: 1857  
|| Cifra 5 se nalazi u zapisu!
```

Primer 2

```
|| INTERAKCIJA SA PROGRAMOM:  
|| Unesite broj: 84  
|| Cifra 5 se ne nalazi u zapisu!
```

[Rešenje 0.10]

Zadatak 0.11 Sa standardnog ulaza unosi se ceo broj. Napisati program koji na standardni izlaz ispisuje odgovor da li je uneti prirodan broj deljiv sumom svojih cifara.

[Rešenje 0.62]

Zadatak 0.12 Napisati program koji učitava ceo neoznačen broj i uklanja sve nule sa desne strane unetog broja. Novodobijeni broj ispisati na standardni izlaz.

Primer 1

```
|| INTERAKCIJA SA PROGRAMOM:  
| Unesite broj: 12000  
| 12
```

Primer 2

```
|| INTERAKCIJA SA PROGRAMOM:  
| Unesite broj: 856  
| 856
```

Primer 3

```
|| INTERAKCIJA SA PROGRAMOM:  
| Unesite broj: 140  
| 14
```

[Rešenje 0.12]

Zadatak 0.13 Napisati program koji učitava neoznačeni ceo broj i transformiše ga tako što svaku parnu cifru u zapisu broja uveća za 1. Novodobijeni broj ispisati na standardni izlaz.

Primer 1

```
|| INTERAKCIJA SA PROGRAMOM:  
| Unesite broj: 2417  
| 3517
```

Primer 2

```
|| INTERAKCIJA SA PROGRAMOM:  
| Unesite broj: 138  
| 139
```

Primer 3

```
|| INTERAKCIJA SA PROGRAMOM:  
| Unesite broj: 59  
| 59
```

[Rešenje 0.13]

Zadatak 0.14 Sa standardnog ulaza unosi se neoznačen ceo broj. Napisati program koji formira i ispisuje broj koji se dobija izbacivanjem svake druge cifre polaznog broja, počevši od krajnje desne cifre.

Primer 1

```
|| INTERAKCIJA SA PROGRAMOM:  
| Unesite broj: 21854  
| 284
```

Primer 2

```
|| INTERAKCIJA SA PROGRAMOM:  
| Unesite broj: 18  
| 8
```

Primer 3

```
|| INTERAKCIJA SA PROGRAMOM:  
|| Unesite broj: 1  
|| 1
```

[Rešenje 0.14]

* **Zadatak 0.15** Sa standardnog ulaza unosi se neoznačen ceo broj. Napisati program koji formira i ispisuje broj koji se dobija izbacivanjem cifara koje su jednake zbiru svojih suseda.

Primer 1

```
|| INTERAKCIJA SA PROGRAMOM:  
|| Unesite broj: 28631  
|| 2631
```

Primer 2

```
|| INTERAKCIJA SA PROGRAMOM:  
|| Unesite broj: 440  
|| 40
```

Primer 3

```
|| INTERAKCIJA SA PROGRAMOM:  
|| Unesite broj: 242  
|| 22
```

[Rešenje 0.15]

* **Zadatak 0.16** Broj je *palindrom* ukoliko se isto čita i sa leve i sa desne strane. Napisati program koji učitava ceo neoznačen broj i proverava da li je učitani broj palindrom.

Primer 1

```
|| INTERAKCIJA SA PROGRAMOM:  
|| Unesite broj: 25452  
|| Broj je palindrom!
```

Primer 2

```
|| INTERAKCIJA SA PROGRAMOM:  
|| Unesite broj: 895  
|| Broj nije palindrom!
```

Primer 3

```
|| INTERAKCIJA SA PROGRAMOM:  
|| Unesite broj: 5  
|| Broj je palindrom!
```

[Rešenje 0.16]

0.1.3 Unos i obrada veće količine podatka (unos i obrada niza brojeva?, nije sjajno zbog nizova)

Zadatak 0.17 Napisati program koji učitava pozitivan ceo broj n , a zatim učitava n celih brojeva i na standardni izlaz ispisuje sumu pozitivnih i sumu negativnih unetih brojeva.

[Rešenje 0.17]

Zadatak 0.18 Sa standardnog ulaza unosi se ceo pozitivan broj n , a potom i n celih brojeva. Izračunati i ispisati zbir onih brojeva koji su neparni i negativni.

Primer 1

```
|| INTERAKCIJA SA PROGRAMOM:  
| Unesite broj n: 5  
| Unesite n brojeva: 1 -5 -6 3 -11  
| -16
```

Primer 2

```
|| INTERAKCIJA SA PROGRAMOM:  
| Unesite broj n: 4  
| Unesite n brojeva: -1 1 0 3  
| -1
```

Primer 3

```
|| INTERAKCIJA SA PROGRAMOM:  
| Unesite broj n: 4  
| Unesite n brojeva: 5 8 13 17  
| 0
```

[Rešenje 0.18]

Zadatak 0.19 Napisati program koji učitava cele brojeve sve dok se ne unese nula. Nakon toga ispisati proizvod onih unetih brojeva koji su pozitivni.

[Rešenje 0.19]

Zadatak 0.20 U prodavnici se nalazi n artikala čije cene su realni brojevi. Napisati program koji učitava n , a potom i cenu svakog od n artikala i određuje i na standardni izlaz ispisuje najmanju cenu.

[Rešenje 0.20]

Zadatak 0.21 Sa standardnog ulaza se unose realni brojevi sve do unosa broja nula 0. Napisati program koji izračunava i ispisuje aritmetičku sredinu unetih brojeva.

Primer 1

```
|| INTERAKCIJA SA PROGRAMOM:  
| Unesite brojeve: 8 5.2 6.11 3 0  
| Aritmeticka sredina: 5.5775
```

[Rešenje 0.22]

Zadatak 0.22 U prodavnici se nalaze artikala čije cene su realni pozitivni brojevi. Cene artikala se unose sa standardnog unosa sve do unosa broja nula 0. Napisati program koji izračunava i ispisuje prosečnu vrednost cena u radnji.

I ovo bi moglo da se preformulise u cene, tj da se sracuna prosečna vrednost cena u radnji. Čak mislim da bi mogli da stavimo dva zadatka, ovaj i jedan sa cenama, a u resenju da se pozovemo samo na resenje ovog zadatka, tako da se vidi da je to u sustini isti problem.

Danijela: dodat još jedan zadatak, u rešenju se pozvati na prethodni.

Danijela: obratiti pažnju da cene mogu biti samo pozitivni brojevi, dok u prethodnom zatku nismo imali takav zahtev – da li menjati prethodni zadatak ili dati rešenje i za ovaj?

Primer 1

```
|| INTERAKCIJA SA PROGRAMOM:  
|| Unesite cene: 8 5.2 6.11 3 0  
|| Aritmeticka sredina: 5.5775
```

[Rešenje 0.22]

Zadatak 0.23 U narednim zadacima se u tekstu kaze da se unosi ceo pozitivan broj a posle se u resenju nigde to ne proverava, niti se koristi tip unsigned. Nesto od toga mora, inae resenje nije dobro.

Danijela: sredicu rešenje u odnosu na ovaj komentar.

Danijela: obratiti pažnju na tekst i Peru – da li nam se sviđa ovako nešto? Ako su cene onda mogu biti samo pozitivni brojevi?

Pera želi da obraduje baku i da joj kupi jedan poklon u radnji. On na raspolaganju ima m novaca. U radnji se nalazi n artikala i zanima ga koliko ima artikala u radnji čija cena je manja ili jednaka m . Napisati program koji pomaže Peri da brzo odrediti broj atikala. Program učitava realan pozitivan broj m , ceo neoznačen broj n i n realnih pozitivnih brojeva različitih od 0. Ispisati koliko artikala ima manju ili jednaku cenu od m . U slučaju greške ispisati odgovarajuću poruku.

Primer 1

```
|| INTERAKCIJA SA PROGRAMOM:  
|| Unesite broj m: 12.37  
|| Unesite broj n: 5  
|| Unesite n brojeva: 11 54.13 -6 13 8  
|| 3
```

Primer 2

```
|| INTERAKCIJA SA PROGRAMOM:  
|| Unesite broj m: 2  
|| Unesite broj n: 4  
|| Unesite n brojeva: -1 11 4.32 3  
|| 1
```

[Rešenje 0.23]

Zadatak 0.24 Sa standardnog ulaza unosi se ceo pozitivan broj n , a potom n celih brojeva. Naći sumu brojeva koji su deljivi sa 5, a nisu deljivi sa 7. U slučaju greške pri unosu podataka ispisati odgovarajuću poruku.

Ukoliko test primer ne moze da stane u midi onda treba da bude maxi, ali mozda bolje skratiti ga u midi.

Danijela: bice promenjeno kad budem sredjivala test primere.

Primer 1

```
|| INTERAKCIJA SA PROGRAMOM:
| Unesite broj n: 5 2 35 5 -175 -20
| -15
```

Primer 2

```
|| INTERAKCIJA SA PROGRAMOM:
| Unesite broj n: -3
| -1
```

Primer 3

```
|| INTERAKCIJA SA PROGRAMOM:
| Unesite broj n: 10 -5 6 175 -20 -25 -8 42 245 1
| 6
| -50
```

Primer 4

```
|| INTERAKCIJA SA PROGRAMOM:
| Unesite broj n: 6 2205 -1904 2 7 -540 5
| -535
```

[Rešenje 0.62]

Zadatak 0.25 Sa standardnog ulaza unosi se ceo broj n , a potom n realnih brojeva. Odrediti koliko puta je prilikom unosa došlo do promene znaka. Ispisati dobijenu vrednost na standardni izlaz.

[Rešenje 0.62]

Zadatak 0.26 Sa standardnog ulaza se unosi ceo pozitivan broj n , a zatim i n celih brojeva. Napisati program koji ispisuje broj sa najvećom cifrom desetica. Ukoliko ima više takvih, ispisati prvi.

Primer 1

```
|| INTERAKCIJA SA PROGRAMOM:
| Unesite broj n: 5
| Unesite n brojeva: 18 365 25 1 78
| 78
```

[Rešenje 0.26]

Zadatak 0.27 Sa standardnog ulaza se unosi ceo pozitivan broj n , a zatim i n celih brojeva. Napisati program koji ispisuje broj sa najvećim brojem cifara. Ukoliko ima više takvih, ispisati prvi.

Primer 1

```
|| INTERAKCIJA SA PROGRAMOM:
|| Unesite broj n: 5
|| Unesite n brojeva: 18 365 25 1 78
|| 365
```

Primer 2

```
|| INTERAKCIJA SA PROGRAMOM:
|| Unesite broj n: 7
|| Unesite n brojeva: 3 892 18 21 639 742 85
|| 892
```

[Rešenje 0.27]

Zadatak 0.28 Sa standardnog ulaza se unosi ceo pozitivan broj n , a zatim i n celih brojeva. Napisati program koji ispisuje broj sa najvećom vodećom cifrom. Vodeća cifra je cifra najveće težine u zapisu broja. Ukoliko ima više takvih, ispisati prvi.

Primer 1

```
|| INTERAKCIJA SA PROGRAMOM:
|| Unesite broj n: 5
|| Unesite n brojeva: 8 964 32 511 27
|| 964
```

Primer 1

```
|| INTERAKCIJA SA PROGRAMOM:
|| Unesite broj n: 3
|| Unesite n brojeva: 41 669 8
|| 8
```

[Rešenje 0.28]

Zadatak 0.29 Sa standardnog ulaza se unose celi pozitivni brojevi n ($n > 1$) i d , a zatim i n celih brojeva. Napisati program koji izračunava koliko ima parova uzastopnih brojeva među unetim brojevima koji se nalaze na rastojanju d . Rastojanje između brojeva je definisano sa $d(x, y) = |y - x|$. Rezultat ispisati na standardni izlaz.

Primer 1

```
|| INTERAKCIJA SA PROGRAMOM:
|| Unesite brojeve n i d: 5 2
|| Unesite n brojeva: 2 3 5 1 -1
|| Broj parova: 2
```

Primer 2

```
|| INTERAKCIJA SA PROGRAMOM:
|| Unesite brojeve n i d: 10 5
|| Unesite n brojeva: -3 6 11 -20 -25 -8 42 37 1 6
|| Broj parova: 4
```

[Rešenje 0.29]

Zadatak 0.30 Vršna su merenja nadmorskih visina na određenom delu teritorije i naučnike zanima razlika između najveće i najmanje nadmorske visine. Napisati program koji učitava n , potom n realnih brojeva koji označavaju nadmorske visine i ispisuje razliku najveće i najmanje nadmorske visine.

Primer 1

```
|| INTERAKCIJA SA PROGRAMOM:  
|| Unesite brojeve: 8 6 5 2 11 7 0  
|| Razlika: 9
```

Primer 2

```
|| INTERAKCIJA SA PROGRAMOM:  
|| Unesite brojeve: 8 -1 8 6 0  
|| Razlika: 9
```

[Rešenje 0.30]

0.1.4 Rad sa karakterima

Zadatak 0.31 Napisati program koji učitava karaktere dok se ne unese karakter tačka i ako je karakter malo slovo, ispisuje odgovarajuće veliko, ako je karakter veliko slovo ispisuje odgovarajuće malo, a u suprotnom ispisuje isti karakter kao i uneti.

[Rešenje 0.31]

Zadatak 0.32 Napisati program koji učitava karaktere sve do kraja ulaza, a potom ispisuje broj velikih slova, broj malih slova, broj cifara, broj belina i zbir unetih cifara.

[Rešenje 0.32]

Zadatak 0.33 Sa standardnog ulaza unosi se ceo pozitivan broj n , a potom i n karaktera. Za svaki od samoglasnika ispisati koliko puta se pojavio među unetim karakterima. Ne praviti razliku između malih i velikih slova.

Primer 1

```
|| INTERAKCIJA SA PROGRAMOM:  
|| Unesite broj n: 5  
|| Unesite n karaktera: u A b a o  
|| Samoglasnik a: 2  
|| Samoglasnik e: 0  
|| Samoglasnik i: 0  
|| Samoglasnik o: 1  
|| Samoglasnik u: 0
```

Primer 2

```
|| INTERAKCIJA SA PROGRAMOM:  
|| Unesite broj n: 7  
|| Unesite n karaktera: j k + E E a e  
|| Samoglasnik a: 1  
|| Samoglasnik e: 3  
|| Samoglasnik i: 0  
|| Samoglasnik o: 0  
|| Samoglasnik u: 0
```

[Rešenje 0.33]

Zadatak 0.34 Sa standardnog ulaza se unosi ceo broj n , a zatim i n karaktera. Napisati program koji proverava da li se od unetih karaktera može napisati reč *Zima*.

Primer 1

```
|| INTERAKCIJA SA PROGRAMOM:
|| Unesite broj n: 4
|| Unestite 1. karakter: +
|| Unestite 2. karakter: o
|| Unestite 3. karakter: Z
|| Unestite 4. karakter: j
|| Ne moze se napisati rec Zima.
```

Primer 2

```
|| INTERAKCIJA SA PROGRAMOM:
|| Unesite broj n: 10
|| Unestite 1. karakter: i
|| Unestite 2. karakter: 9
|| Unestite 3. karakter: 0
|| Unestite 4. karakter: p
|| Unestite 5. karakter: a
|| Unestite 6. karakter: Z
|| Unestite 7. karakter: o
|| Unestite 8. karakter: m
|| Unestite 9. karakter: M
|| Unestite 10. karakter: -
|| Moze se napisati rec Zima.
```

[Rešenje 0.34]

0.1.5 Računanje sume i proizvoda

Zadatak 0.35 Prekoracenje se javlja mnooogo ranije. I ovo je jedan od zadataka koji imamo u funkcijama. Napisati program koji učitava ceo pozitivan broj i i izračunava njegov faktorijel. U slučaju neispravnog unosa ispisati odgovarajuću poruku. UPUTSTVO: Obratiti pažnju da počev od broja 23 dolazi do prekoračenja prilikom računanja faktorijela.

[Rešenje 0.35]

Zadatak 0.36 Sa standradnog ulaza unose se realan broj x i ceo neoznačen broj n . Napisati program koji izračunava n -ti stepen broja x , tj. x^n .

Primer 1

```
|| INTERAKCIJA SA PROGRAMOM:
|| Unesite redom brojeve x i n: 4 3
|| 64.00000
```

Primer 2

```
|| INTERAKCIJA SA PROGRAMOM:
|| Unesite redom brojeve x i n: 5.8 5
|| 6563.56768
```

Primer 3

```
|| INTERAKCIJA SA PROGRAMOM:
|| Unesite redom brojeve x i n: 11.43 0
|| 1.00000
```

[Rešenje 0.36]

Zadatak 0.37 Sa standradnog ulaza unose se realan broj x i ceo broj n . Napisati program koji izračunava n -ti stepen broja x .

Primer 1

```
|| INTERAKCIJA SA PROGRAMOM:  
| Unesite redom brojeve x i n: 2 -3  
| 0.125
```

Primer 2

```
|| INTERAKCIJA SA PROGRAMOM:  
| Unesite redom brojeve x i n: -3 2  
| 9.000
```

[Rešenje ??]

Zadatak 0.38 Napisati program koji učitava ceo pozitivan broj n i ispisuje vrednost sume kubova brojeva od 1 do n , odnosno $s = 1 + 2^3 + 3^3 + \dots + n^3$. U slučaju greške pri unosu podataka ispisati odgovarajuću poruku.

[Rešenje 0.39]

Zadatak 0.39 Napisati program koji učitava ceo pozitivan broj n i ispisuje sumu kubova, $s = 1 + 2^3 + 3^3 + \dots + k^3$, za svaku vrednost $k = 1, \dots, n$. U slučaju greške pri unosu podataka ispisati odgovarajuću poruku.

[Rešenje 0.39]

Zadatak 0.40 Sa standardnog ulaza unose se realan broj x i ceo neoznačen broj n . Napisati program koji izračunava i na standardni izlaz ispisuje sumu $S = x + 2 \cdot x^2 + 3 \cdot x^3 + \dots + n \cdot x^n$.

Primer 1

```
|| INTERAKCIJA SA PROGRAMOM:  
| Unesite redom brojeve x i n: 2 3  
| S=34.000000
```

Primer 2

```
|| INTERAKCIJA SA PROGRAMOM:  
| Unesite redom brojeve x i n: 1.5 5  
| S=74.343750
```

[Rešenje 0.40]

Zadatak 0.41 Sa standardnog ulaza unose se realan broj x i ceo neoznačen broj n . Napisati program koji izračunava i na standardni izlaz ispisuje sumu $S = 1 + \frac{1}{x} + \frac{1}{x^2} + \dots + \frac{1}{x^n}$.

Primer 1

```
|| INTERAKCIJA SA PROGRAMOM:  
| Unesite redom brojeve x i n: 2 4  
| S=1.937500
```

Primer 2

```
|| INTERAKCIJA SA PROGRAMOM:  
| Unesite redom brojeve x i n: 1.8 6  
| S=2.213249
```

[Rešenje 0.41]

*** Zadatak 0.42** Mislila sam da se tačnost ϵ odnosi na to da je razlika dva uzastopna člana manja od ϵ a ne da je sam član manji od ϵ ? Nisam sigurna, ali možda treba proveriti ili preformulisati zadatak tako da se ne definiše ovaj pojam. Napisati program koji učitava realane brojeve x i ϵ i sa zadatom tačnošću ϵ izračunava i na standardni izlaz ispisuje sumu $S = 1 + x + \frac{x^2}{2!} + \frac{x^3}{3!} + \dots$. Izračunati sumu u odnosu na tačnost ϵ znači uporediti poslednji član sume sa ϵ i ukoliko je taj poslednji član manji od ϵ prekinuti dalja izračunavanja. UPUTSTVO: Prilikom računanja sume koristiti prethodni izračunati član sume u računanju sledećeg člana sume. Naime, ako je izračunat član sume $\frac{x^n}{n!}$ na osnovu njega se lako može dobiti član $\frac{x^{n+1}}{(n+1)!}$. Nikako ne računati stepen i faktorijel odvojeno zbog neefikasnosti takvog rešenja i zbog mogućnosti prekoračenja.

Primer 1

```
|| INTERAKCIJA SA PROGRAMOM:
|| Unesite x: 2
|| Unesite tacnost eps: 0.001
|| S=7.388713
```

Primer 2

```
|| INTERAKCIJA SA PROGRAMOM:
|| Unesite x: 3
|| Unesite tacnost eps: 0.01
|| S=20.079666
```

[Rešenje 0.42]

*** Zadatak 0.43** Napisati program koji učitava realane brojeve x i ϵ i sa zadatom tačnošću ϵ izračunava i na standardni izlaz ispisuje sumu $S = 1 - x + \frac{x^2}{2!} - \frac{x^3}{3!} + \frac{x^4}{4!} - \frac{x^5}{5!} \dots$. NAPOMENA: Voditi računa o efikasnosti rešenja i o mogućnosti prekoračenja.

Primer 1

```
|| INTERAKCIJA SA PROGRAMOM:
|| Unesite x: 3
|| Unesite tacnost eps: 0.001
|| S=0.049997
```

Primer 2

```
|| INTERAKCIJA SA PROGRAMOM:
|| Unesite x: 3.14
|| Unesite tacnost eps: 0.01
|| S=0.049072
```

[Rešenje 0.43]

Zadatak 0.44 Napisati program koji učitava realan broj x i prirodan broj n izračunava sumu $S = (1 + \cos(x)) \cdot (1 + \cos(x^2)) \cdot \dots \cdot (1 + \cos(x^n))$. NAPOMENA: Voditi računa o efikasnosti rešenja.

[Rešenje 0.62]

*** Zadatak 0.45** Napisati program koji učitava ceo neoznačen broj n , a na standardni izlaz ispisuje vrednost razlomka

$$1 + \frac{1}{2 + \frac{1}{3 + \frac{1}{4 + \frac{1}{\dots + \frac{1}{(n-1) + \frac{1}{n}}}}}}.$$

[Rešenje 0.62]

* **Zadatak 0.46** Napisati program koji računa sumu

$$1 - \frac{x^2}{2!} + \frac{x^4}{4!} - \dots + (-1)^n \frac{x^{2n}}{(2n)!}.$$

za unete cele brojeve x i n . NAPOMENA: *Voditi računa o efikasnosti rešenja i o mogućnosti prekoračenja.*

[Rešenje 0.62]

* **Zadatak 0.47** Sa standardnog ulaza unosi se ceo pozitivan broj n veći od 0. Napisati program koji računa proizvod

$$S = (1 + \frac{1}{2!})(1 + \frac{1}{3!}) \dots (1 + \frac{1}{n!}).$$

U slučaju greške pri unosu podataka ispisati odgovarajuću poruku. NAPOMENA: *Voditi računa o efikasnosti rešenja i o mogućnosti prekoračenja.*

Primer 1

```
|| INTERAKCIJA SA PROGRAMOM:
|| Unesite broj n: 5
|| 1.838108
```

Primer 2

```
|| INTERAKCIJA SA PROGRAMOM:
|| Unesite broj n: 7
|| 1.841026
```

Primer 3

```
|| INTERAKCIJA SA PROGRAMOM:
|| Unesite broj n: 0
|| -1
```

Primer 4

```
|| INTERAKCIJA SA PROGRAMOM:
|| Unesite broj n: 10
|| 1.841077
```

[Rešenje 0.62]

* **Zadatak 0.48** Sa standardnog ulaza unosi se ceo pozitivan neparan broj n . Napisati program koji za uneto n izračunava:

$$S = 1 \cdot 3 \cdot 5 - 1 \cdot 3 \cdot 5 \cdot 7 + 1 \cdot 3 \cdot 5 \cdot 7 \cdot 9 - 1 \cdot 3 \cdot 5 \cdot 7 \cdot 9 \cdot 11 + \dots (-1)^{\frac{n-1}{2}+1} \cdot 1 \cdot 3 \cdot \dots \cdot n.$$

U slučaju greške pri unosu podataka ispisati odgovarajuću poruku. NAPOMENA: *Voditi računa o efikasnosti rešenja i o mogućnosti prekoračenja.*

Primer 1

```
|| INTERAKCIJA SA PROGRAMOM:
|| Unesite broj n: 9
|| 855
```

Primer 2

```
|| INTERAKCIJA SA PROGRAMOM:
|| Unesite broj n: 11
|| -9540
```

Primer 3

```
|| INTERAKCIJA SA PROGRAMOM:
|| Unesite broj n: 20
|| -1
```

Primer 4

```
|| INTERAKCIJA SA PROGRAMOM:
|| Unesite broj n: -3
|| -1
```

[Rešenje 0.62]

Zadatak 0.49 Sa standardnog ulaza unose se realni brojevi x i a i ceo pozitivan broj n veći od 0. Napisati program koji izračunava:

$$((\dots(\underbrace{((x+a)^2+a)^2+a)^2+\dots a}_n)^2).$$

U slučaju greške pri unosu podataka ispisati odgovarajuću poruku.

Primer 1

```
|| INTERAKCIJA SA PROGRAMOM:
|| Unesite broj n: 3.2 0.2 5
|| 367940960.000000
```

Primer 2

```
|| INTERAKCIJA SA PROGRAMOM:
|| Unesite broj n: 2 1 3
|| 101.000000
```

Primer 3

```
|| INTERAKCIJA SA PROGRAMOM:
|| Unesite broj n: 2.6 0.3 3
|| 76.164085
```

Primer 4

```
|| INTERAKCIJA SA PROGRAMOM:
|| Unesite broj n: 5.4 7 -2
|| -1
```

[Rešenje 0.62]

0.1.6 Dvostruka petlja i ispisivanje slike

Zadatak 0.50 Sa standardnog ulaza unosi se neoznačen broj n . Napisati program koji za uneto n zvezdicama iscrtava

a) kvadrat stranice n sastavljen od zvezdica.

Primer 1

```
|| INTERAKCIJA SA PROGRAMOM:
|| Unesite broj n: 3
|| ***
|| ***
|| ***
```

b) rub kvadrata dimenzije n .

Primer 1

```
|| INTERAKCIJA SA PROGRAMOM:
|| Unesite broj n: 5
|| *****
|| * *
|| * *
|| * *
|| *****
```

Primer 2

```
|| INTERAKCIJA SA PROGRAMOM:
|| Unesite broj n: 2
|| **
|| **
```

c) rub kvadrata dimenzije n koji i na glavnoj dijagonali ima zvezdice.

Primer 1

```
|| INTERAKCIJA SA PROGRAMOM:
|| Unesite broj n: 5
|| *****
|| ** *
|| * * *
|| * **
|| *****
```

[Rešenje 0.62]

* **Zadatak 0.51** Napisati program koji za uneti ceo broj n zvezdicama iscrtava slovo X dimenzije n .

Primer 1

```
|| INTERAKCIJA SA PROGRAMOM:
|| Unesite broj n: 5
|| * *
|| * *
|| *
|| * *
|| * *
|| * *
```

Primer 2

```
|| INTERAKCIJA SA PROGRAMOM:
|| Unesite broj n: 3
|| * *
|| *
|| * *
```

[Rešenje 0.52]

* **Zadatak 0.52** Napisati program koji za uneti ceo broj n korišćenjem znaka + iscrtava veliko + dimenzije n .

Primer 1

```
|| INTERAKCIJA SA PROGRAMOM:
|| Unesite broj n: 5
|| +
|| +
|| +++++
|| +
|| +
```

Primer 2

```
|| INTERAKCIJA SA PROGRAMOM:
|| Unesite broj n: 3
|| +
|| +++
|| +
```

[Rešenje 0.52]

Zadatak 0.53 Napisati program koji učitava ceo neoznačen broj n , a potom iscrtava

- a) pravougli trougao sastavljen od zvezdica. Kateta trougla je dužine n , a prav ugao se nalazi u gornjem levom uglu slike.

Primer 1

```
|| INTERAKCIJA SA PROGRAMOM:
|| Unesite broj n: 3
|| ***
|| **
|| *
```

- b) pravougli trougao sastavljen od zvezdica. Kateta trougla je dužine n , a prav ugao se nalazi u donjem levom uglu slike.

Primer 1

```
|| INTERAKCIJA SA PROGRAMOM:
|| Unesite broj n: 3
|| *
|| **
|| ***
```

- c) trougao sastavljen od zvezdica. Trougao se dobija spajanjem dva pravougla trougla čija kateta je dužine n , pri čemu je prav ugao prvog trougla u njegovom donjem levom uglu, dok je prav ugao drugog trougla u njegovom gornjem levom uglu, a spajanje se vrši po horizontalnoj kateti.

Primer 1

```
|| INTERAKCIJA SA PROGRAMOM:
|| Unesite broj n: 3
|| *
|| **
|| ***
|| **
|| *
||
```

- d) rub jednakokrakog pravouglog trougla čije su katete dužine n . Program učitava karakter c i taj karakter koristi za iscrtavanje ruba trougla.

Primer 1

```
|| INTERAKCIJA SA PROGRAMOM:
|| Unesite broj n: 4
|| Unesite karakter c: *
|| *
|| **
|| * *
|| ****
||
```

Primer 2

```
|| INTERAKCIJA SA PROGRAMOM:
|| Unesite broj n: 5
|| Unesite karakter c: +
|| +
|| ++
|| + +
|| + +
|| +++++
||
```

[Rešenje 0.62]

Zadatak 0.54 Napisati program koji učitava ceo broj n , a potom iscrtava

- a) jednakostranični trougao stranice n koji je sastavljen od zvezdica.

Primer 1

```
|| INTERAKCIJA SA PROGRAMOM:
|| Unesite broj n: 3
|| *
|| ***
|| *****
||
```

- b) trougao koji se dobija spajanjem dva jednakostranični trougla stranice n koji su sastavljeni od zvezdica.

Primer 1

```
|| INTERAKCIJA SA PROGRAMOM:
|| Unesite broj n: 3
|| *
|| ***
|| *****
|| ***
|| *
||
```

c) rub jednakostraničnog trougla čija stranica je dužine n .

Primer 1

```
INTERAKCIJA SA PROGRAMOM:
Unesite broj n: 3
*
* *
* * *
```

d) sliku koja se dobija spajanjem dva jednakostranična trougla čija stranica je dužine n . Iscrtavati samo rub trouglova.

Primer 1

```
INTERAKCIJA SA PROGRAMOM:
Unesite broj n: 3
*
* *
* * *
* *
*
```

[Rešenje 0.62]

* **Zadatak 0.55** Napisati program koji za uneti ceo broj n iscrtava strelice dimenzije n .

Primer 1

```
INTERAKCIJA SA PROGRAMOM:
Unesite broj n: 3
*
*
***
*
*
```

Primer 2

```
INTERAKCIJA SA PROGRAMOM:
Unesite broj n: 5
*
*
*
*
*****
*
*
*
*
*
```

[Rešenje 0.55]

* **Zadatak 0.56** Napisati program koji učitava ceo broj n , i iscrtava sliku koja se dobija na sledeći način: u prvom redu je jedna zvezdica, u drugom redu su dve zvezdice razdvojene razmakom, treći red je sastavljen od zvezdica i iste je dužine kao i drugi red, četvrti red se sastoji od tri zvezdice razdvojene razmakom, a peti red je sastavljen od zvezdica i iste je dužine kao i četvrti red itd. Ukupna visina slike je n .

Primer 1

```
INTERAKCIJA SA PROGRAMOM:
Unesite broj n: 7
*
* *
***
* * *
****
* * * *
*****
```

[Rešenje 0.62]

**** Zadatak 0.57** Sa standardnog ulaza unose se neoznačeni celi brojevi m i n . Napisati program koji iscrtava jedan do drugog stranice n kvadrata čija je svaka strana sastavljena od m zvezdica razdvojenih prazninom.

Tekst nije u skladu sa slikom jer nije jasno da se crtaju samo rubovi a ne popunjeni kvadrati.

Danijela: Da li je sada jasnije?

Primer 1

```
INTERAKCIJA SA PROGRAMOM:
Unesite broj n: 5 3
* * * * *
*   *   *   *
*   *   *   *
*   *   *   *
* * * * *
```

Primer 2

```
INTERAKCIJA SA PROGRAMOM:
Unesite broj n: 4 4
* * * * *
*   *   *   *
*   *   *   *
* * * * *
```

[Rešenje 0.62]

*** Zadatak 0.58** Sa standardnog ulaza unosi se ceo neoznačen broj n . Napisati program koji štampa romb sastavljen od minusa u pravougaoniku sastavljenom od zvezdica.

Primer 1

```
INTERAKCIJA SA PROGRAMOM:  
Unesite broj n: 6  
*****  
*****--*****  
*****-----*****  
***------***  
**-----**  
*------*  
*------**  
****-----**  
***------***  
*****-----*****  
*****--*****  
*****
```

Primer 2

```
INTERAKCIJA SA PROGRAMOM:
Unesite broj n: 2
****
*--*
****
```

[Rešenje 0.62]

Zadatak 0.59 Napisati program koji učitava ceo broj n ($n \geq 2$) i koji na standardni izlaz iscrta sliku kuće sa krovom: kuća je kocka stranice n , a krov jednakostranični trougao stranice n .

Primer 1

```
INTERAKCIJA SA PROGRAMOM:
Unesite broj n: 5

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

[Rešenje 0.62]

Zadatak 0.60 Sa standardnog ulaza učitava se ceo neoznačen broj n . Napisati program koji za uneto n iscrtaava pravougli „trougao” sačinjen od „koordinata” svojih tačaka. „Koordinata” tačke je oblika (i, j) pri čemu $i, j = 0, \dots, n$. Pravougao se nalazi u gornjem levom uglu slike i njegova koordinata je $(0, 0)$. Koordinata i se uvećava po vrsti, a koordinata j po koloni, pa je zato koordinata tačke koja je ispod tačke $(0, 0)$ jednaka $(1, 0)$, a koordinata tačke koja je desno od tačke $(0, 0)$ jednaka $(0, 1)$.

Ovo treba preformulisati jer je bez test primera skroz nejasno. U test prime-
rima negde ima blanko posle zareza, negde nema, i to treba ujednaciti.

Mene ovaj zadatak zbunjuje i ne sviđa mi se. Problem su mi koordinate koje se broje nekako cudno i to od broja 1 a ne od nule. Nije mi jasno zasto u temenu pravog ugla ne bi bila koordinata (0,0)?

Danijela: Izmenila sam test primere i tekst, ali se slazem da zadatak nije nesto, mozemo ga izbrisati.

Primer 1

```
|| INTERAKCIJA SA PROGRAMOM:  
| Unesite broj n: 1  
| (0,0)
```

Primer 2

```
|| INTERAKCIJA SA PROGRAMOM:  
| Unesite broj n: 2  
| (0,0) (0,1)  
| (1,0)
```

Primer 3

```
|| INTERAKCIJA SA PROGRAMOM:  
| Unesite broj n: 3  
| (0,0) (0,1) (0,2)  
| (1,0) (1,1)  
| (2,0)
```

Primer 4

```
|| INTERAKCIJA SA PROGRAMOM:  
| Unesite broj n: 4  
| (0,0) (0,1) (0,2) (0,3)  
| (1,0) (1,1) (1,2)  
| (2,0) (2,1)  
| (3,0)
```

[Rešenje 0.62]

* **Zadatak 0.61** Sa standardnog ulaza unosi se ceo pozitivan broj n . Napisati program koji ispisuje brojeve od 1 do n , zatim od 2 do $n - 1$, 3 do $n - 2$, itd. Ispis se završava kada nije moguće ispisati ni jedan broj. Za neispravan unos, program ispisuje odgovarajuću poruku.

Primer 1

```
|| INTERAKCIJA SA PROGRAMOM:  
| Unesite broj n: 5  
| 1 2 3 4 5 2 3 4 3
```

Primer 2

```
|| INTERAKCIJA SA PROGRAMOM:  
| Unesite broj n: -4  
| -1
```

Primer 3

```
|| INTERAKCIJA SA PROGRAMOM:  
| Unesite broj n: 5  
| 1 2 3 4 5 6 7 2 3 4 5 6 3 4 5 4
```

Primer 4

```
|| INTERAKCIJA SA PROGRAMOM:  
| Unesite broj n: 3  
| 1 2 3 2
```

[Rešenje 0.62]

* **Zadatak 0.62** Napisati program koji učitava ceo pozitivan broj n i ispisuje sve brojeve od 1 do n , zatim svaki drugi broj od 1 do n , zatim svaki treći broj od 1 do n itd., završavajući sa svakim n -tim (tj. samo sa 1). U slučaju greške pri unosu podataka odštampati odgovarajuću poruku.

Primer 1

```
INTERAKCIJA SA PROGRAMOM:  
Unesite broj n: 3  
1 2 3  
1 3  
1
```

Primer 2

```
INTERAKCIJA SA PROGRAMOM:  
Unesite broj n: 1  
1
```

Primer 3

```
INTERAKCIJA SA PROGRAMOM:  
Unesite broj n: 7  
1 2 3 4 5 6 7  
1 3 5 7  
1 4 7  
1 5  
1 6  
1 7  
1
```

Primer 4

```
INTERAKCIJA SA PROGRAMOM:  
Unesite broj n: -23
```

[Rešenje 0.62]

0.2 Rešenja

Rešenje 0.1

```
1 #include <stdio.h>  
2  
3 int main()  
4 {  
5     /* Promenljiva i kontrolise koliko puta ce se petlja izvesti.  
6      Njenu pocetnu vrednost postavljamo na 0 jer se u pocetku petlja  
7      nije ni jednom izvela. */  
8     int i = 0;  
9  
10    /* Pre ulaska u telo petlje proverava se da li je  
11     ispunjen uslov petlje.  
12     */  
13    while(i < 5)  
14    {  
15        /* Ukoliko uslov petlje jeste ispunjen ulazimo u telo petlje. */  
16        /* Ispisujemo trazeni tekst. */  
17        printf("Mi volimo da programiramo.\n");  
18  
19        /* Uvecavamo promenljivu za jedan jer smo jednom prosli kroz  
20        petlju. */  
21        i++;  
22  
23        /* Nakon poslednje naredbe tela petlje ponovo se vracamo na  
24        ispitivanje uslova petlje.
```

```

22     Ako ovu vrednost ne menjamo dobicemo petlju koja se izvsava
    beskonacno. */
    }
24     return 0;
26 }

```

Rešenje 0.2

```

#include<stdio.h>
2
int main()
4 {
    /* Promenljiva i kontrolise koliko puta ce se petlja izvsiti.
    Najcesce ovakvu promenljivu nazivamo "brojac". */
6     int i=0;
    /* Promenljiva koja oznacava koliko puta cemo ispisati trazeni
    tekst. */
8     int n;

10     printf("Unesite ceo broj: ");
    scanf("%d", &n);
12

    /* Pre ulaska u telo petlje proverava se da li je ispunjen uslov
    petlje. */
14     while (i<n)
    {
16         printf("Mi volimo da programiramo.\n");
        i++;
18     }
    return 0;
20 }

```

Rešenje 0.3

```

1  #include <stdio.h>
    #include <stdlib.h>
3
int main()
5 {
    /* Promenljivu x koristimo u dve svrhe. Prvo, ova promenljiva
    kontrolise koliko puta se petlja izvsila.
7     Drugo, ovu promenljivu koristimo za ispis potrebnih vrednosti.
    */
    int x;
9     /* Promenljiva n se unosi i odredjuje koliko brojeva ispisujemo.
    */
    int n;
11

```

```

13     printf("Unesi pozitivan ceo broj: ");
    scanf("%d", &n);

15     /* U slucaju neispravnih podataka ispisujemo odgovarajucu poruku
    i izlazimo iz programa. */
17     if (n < 0)
    {
19         printf("Neispravan unos. Promenljiva mora biti pozitivna!\n");
        exit(EXIT_FAILURE);
21     }

23     /* Ispis pocinjemo od 0, zato promenljivu x postavljamo na 0. */
    x=0;
25     while (x<= n)
    {
27         /* Ispisujemo broj. */
        printf("%d\n", x);
29         /* Uvecavamo promenljivu za jedan jer smo broj ispisali i sada
        zelimo da ispisemo sledeci broj. */
        x++;
31     }
    return 0;
33 }

```

Rešenje 0.4

```

1  /* Resenje pod a). */

3  #include <stdio.h>
    #include <stdlib.h>

5

7  int main()
    {
9      /* Promenljive koje oznacavaju granice intervala. */
        int n,m;
        /* Promenljiva koja oznacava trenutno ispisani broj intervala. */
11       int i;

13       printf("Unesi dva cela broja: ");
        scanf("%d%d",&n,&m);

15

17       if (m < n)
        {
            printf("Neispravan unos. Nisu dobro zadate granice intervala!\n"
            );
19             exit(EXIT_FAILURE);
        }

21

        /* Na pocetku ispisujemo prvi broj intervala, a to je n. */
23       i=n;
        /* uslov petlje se proverava pre ulaska u telo petlje */

```



```

25     while (i<=m)
26     {
27         printf("%d ", i);
28         i++;
29     }
30
31     printf("\n");
32
33     return 0;
34 }

```

```

1  /* Resenje pod b). */
2
3  #include <stdio.h>
4  #include <stdlib.h>
5
6  int main()
7  {
8
9      /* Promenljive koje oznacavaju granice intervala. */
10     int n,m;
11     /* Promenljiva koja oznacava trenutno ispisani broj intervala. */
12     int i;
13
14     printf("Unesi dva cela broja: ");
15     scanf("%d%d",&n,&m);
16
17     if (m < n)
18     {
19         printf("Neispravan unos. Nisu dobro zadate granice intervala!\n");
20         exit(EXIT_FAILURE);
21     }
22
23     /* naredba i=n se izvorsava jednom, pre prve
24     iteracije */
25     for(i=n;i<=m;i++) /* uslov petlje i<=m se proverava pre svake
26     iteracije */
27         printf("%d ", i); /* naredba i++ se izvorsava nakon svake
28     iteracije */
29
30     printf("\n");
31
32     return 0;
33 }

```

```

1  /* Resenje pod c). */
2

```

```

#include <stdio.h>
#include <stdlib.h>

int main()
{
    /* Promenljive koje oznacavaju granice intervala. */
    int n,m;
    /* Promenljiva koja oznacava trenutno ispisani broj intervala. */
    int i;

    printf("Unesi dva cela broja: ");
    scanf("%d%d",&n,&m);

    if (m < n)
    {
        printf("Neispravan unos. Nisu dobro zadate granice intervala!\n");
        exit(EXIT_FAILURE);
    }

    /* Uslov petlje se proverava na kraju svake iteracije. */
    /* Zbog toga se do while petlja izvršava bar jednom, čak i u
       slucaju
       da uslov petlje nikada nije ispunjen. */
    i=n;
    do
        /* Petlja se zapocinje bez provere uslova. */
    {
        printf("%d ",i); /* Stampa se vrednost promenljive i. */
        i++;             /* Uvecava se vrednost promenljive i. */
    }
    while(i<=m);         /* Proverava se uslov i ukoliko je ispunjen,
                           nastavlja se sa sledecom iteracijom. */
                           /* U suprotnom, petlja se završava i program
                           se nastavlja od prve naredbe koja sledi za petljom. */
    printf("\n");

    return 0;
}

```

Rešenje 0.5

```

#include <stdio.h>
#include <stdlib.h>

int main()
{
    /* Pamtimo uzastopna dva Fibonacijeva broja i na osnovu njih
       racunamo sledeci. */

```

```

8  /* Promenljive prvi i drugi su brojevi koje pamtimo i na osnovu
   /* njih racunamo treci. */
10 /* Na osnovu teksta zadatka, promenljive prvi i drugi postavljamo
   na 1. */
   int prvi = 1;
10  int drugi = 1;
   int treci;
12 /* Promenljiva pozicija je podatak koji ucitavamo i odnosi se na
   poziciju u Fibonaccijevom nizu
   za koju treba izracunati vrednost. */
14 int pozicija;
   /* Promenljiva i oznacava do koje pozicije smo izracunali vrednosti
   . Kako imamo prve dve
16 vrednosti, ovu promenljivo postavljamo na 2. */
   int i = 2;

18 printf("Unesite poziciju u Fibonaccijevom nizu: ");
20 scanf("%d", &pozicija);

22 /* Pozicija ne moze biti 0 i ne moze biti negativan broj. */
   if (pozicija < 1)
24 {
       printf("Neispravan unos. Pozicija u Fibonaccijevom nizu mora biti
       pozitivan broj koji nije 0!\n");
26       exit(EXIT_FAILURE);
   }

28 while(i < pozicija)
30 {
       /* Na osnovu dva uzastopna racunamo treci. */
32       treci = prvi + drugi;

34       /* Potom razmenjujemo vrednosti. Uzastopna dva koja pamtimo
       postaju
       sledeca uzastopna dva broja Fibonaccijevog niza. */
36       prvi = drugi;
       drugi = treci;

38       /* Prelazimo na racunanje sledeceg broja na sledecoj poziciji. */
40       i++;
   }

42 printf("Trazeni broj je: %d\n", drugi);
44 return 0;
46 }

```

Rešenje 0.6

```

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

```

```

3 {
   int a0;
5   int an, an1;

7   printf("Unesi pocetni clan niza brojeva:");
   scanf("%d", &a0);

9   if (a0 > 0)
11  {
      printf("%d\n", a0);

13      an = a0;
15      while (an != 1)
      {
17          if (an % 2) /* Ukoliko je vrednost izraza an%2 razlicita od nule,
              */
              { /* izraz se tumaci kao tacan i izvrsavaju se naredbe
19                  iz if grane. */
                  an1 = (3 * an + 1) / 2;
              }
21          else /* U suprotnom, ukoliko je vrednost izraza an%2 jednaka
              nuli, izraz */
              { /* se tumaci kao netacan i izvrsavaju se naredbe iz else
23                  grane. */
                  an1 = an / 2;
              }
25          printf("%d\n", an1);
          an = an1;
27      }
   }
29   else
       printf("Nekorektan unos. Broj mora biti pozitivan.\n");
31   return 0;
33 }

```

Rešenje 0.62

Rešenje 0.8

```

1  /*
   Napisati program koji za uneti ceo broj ispisuje njegove cifre
3  u obrnutom poretku.
   */
5
   #include <stdio.h>
7   #include <stdlib.h>
   int main()
9   {
       int x;

```

```

11  char cifra;
    printf("Unesi ceo broj:");
13  scanf("%d", &x);

15  x = abs(x); /* pretvaranje u apsolutnu vrednost se vrši za slučaj
               kada je unet          negativan broj kako bismo osigurali da će nam
               izdvojene cifre      biti pozitivne
17               */
19
    while(x>0)
21  {
        cifra=x%10;          /* izdvajamo poslednju cifru broja x */
23  printf("%d\\n", cifra);
        x/=10;              /* ako je npr x=1582, x%10 će biti 2,
25                             a x/10 će biti 158;
                             npr x=5, x%10 će biti 5
27                             a x/10 će biti 0 */
29  }

    return 0;
31 }

```

Rešenje 0.9

```

1  /*
   Napisati program koji ispisuje sve prave delioce unetog pozitivnog
   celog broja.
3
   */
5
   #include<stdio.h>
7   #include<math.h>
   int main()
9   {
       int x;
11      int i;

13      printf("Unesi x>0:");
       scanf("%d", &x);

15      if (x<=0)
17      {
           printf("Neispravan unos\\n");
19  return -1;
       }

21      /* 1. nacin */
23      printf("----- 1. nacin -----\\n");
       for(i=2;i<x;i++)

```

```

25 {
26     printf("proveravam za %d...\n",i);
27     if (x%i==0)
28         printf("\t delilac:%d \n",i);
29 }
30 /* 2. nacin (brzi) */
31 printf("----- 2. nacin ----- \n");
32 for(i=2;i<=sqrt(x);i++)
33 {
34     printf("proveravam za %d...\n",i);
35     if (x%i==0)
36         if (i==x/i) /* u slucaju kada je delilac koren broja, npr 4
37            za 16, ispisujemo ga jednom */
38             printf("\t delilac:%d \n",i);
39         else /* u suprotnom, npr 2 za 16, ispisujemo i 2 i 8
40             */
41             printf("\t delioci:%d %d \n",i,x/i);
42     }
43 return 0;
44 }

```

Rešenje 0.10

```

1  /* Sa standardnog ulaza unosi se ceo neoznaceni broj. Napisati program
2     koji
3     proverava i ispisuje da li se cifra 5 nalazi u njegovom zapisu ili ne
4     . */
5
6  #include <stdio.h>
7
8  int main(){
9     int n, cifra;
10    int indikator=0;
11
12    /* Ucitavamo broj */
13    printf("Unesite broj: ");
14    scanf("%d", &n);
15
16    /* Sve dok imamo cifara u zapisu broja */
17    while(n>0){
18
19        /* Izdvajamo poslednju cifru broja */
20        cifra=n%10;
21
22        /* Proveravamo da li je bas ona jednaka broju 5 */
23        if(cifra==5){
24            /* Ako jeste postavljamo indikator na vrednost 1 tako da
25               znamo da smo
26               * pronasli peticu i prekidamo sa izvršavanjem petlje */
27            indikator=1;
28            break;
29        }
30    }
31
32    if(indikator==1)
33        printf("Broj sadrzi cifru 5\n");
34    else
35        printf("Broj ne sadrzi cifru 5\n");
36
37    return 0;
38 }

```

```

27     }
    /* Ako izdvojena cifra nije jednaka broju 5, broj delimo sa 10
    kako bi
        mogli da izdvojimo i preostale cifre broja na isti nacin */
29     n=n/10;
    }

31     /* Ispisujemo rezultat */
33     if(indikator==0){
        printf("Cifra 5 se ne nalazi u zapisu!\n");
35     }
    else{
37         printf("Cifra 5 se nalazi u zapisu!\n");
    }

39     return 0;
41 }

```

Rešenje 0.62

Rešenje 0.12

```

1  /* Napisati program koji unetom broju uklanja nule sa desne strane.
   Novodobijeni
   broj ispisati na standardni izlaz. */
3
   #include <stdio.h>
5
   int main(){
7       int n;

9       /* Ucitavamo broj */
       printf("Unesite broj: ");
11      scanf("%d", &n);

13      if(n==0){
          printf("0\n");
15      }
      else{
17          /* Sve dok je poslednja cifra u zapisu broja n nula */
          while(n%10==0){
19              /* Broj delimo sa 10 tj. uklanjamo mu nulu sa kraja */
              n=n/10;
21          }

23          /* Ispisujemo rezultat */
          printf("%d\n", n);
25      }

27

```

```
    return 0;
29 }
```

Rešenje 0.13

```
1  #include <stdio.h>
3  int main() {
5      unsigned int x;
   int pozicija;    // da li se radi o cifri jedinici, desetici,
   stotini itd...
7      int cifra;    // trenutna izdvojena cifra iz broja x
   unsigned int y;  // broj dobijen nakon transformacije
9
   printf("Unesite broj: ");
11  scanf("%d", &x);
13
   if(x > 0) {
15
       /* Posto pocinjemo sa izdvajanjem cifara od cifre jedinica,
17        postavljamo tezinu (stepen) pozicije na 1 */
       pozicija = 1;
19       y = 0;
21
       /* Sve dok imamo cifara u zapisu broja */
       while(x > 0) {
23
           /* Izdvajamo poslednju cifru iz zapisa */
25           cifra = x % 10;
27
           /* Proveravamo da li je cifra parna */
           if(cifra % 2 == 0){
29               /* I ako jeste, uvecavamo je */
               cifra++;
31
           }
33
           /* Novi broj formiramo tako sto izdvojenu cifru pomnozimo
           odgovarajucom
35               tezinom (stepenom) pozicije */
           y += cifra*pozicija;
37
           /* Pripremamo broj za izdvajanje naredne cifre */
39           x /= 10;
41
           /* I uvecavamo tezinu (stepen) pozicije */
           pozicija *= 10;
43       }
   }
```



```

45     /* Ispisujemo izracunatu vrednost */
    printf("%d\n", y);
47 }
    else
49     printf("Nekorektan unos.\n");

51     return 0;
}

```

Rešenje 0.14

```

1  /* Sa standardnog ulaza unosi se neoznaceni ceo broj. Napisati program
    koji
    formira i ispisuje broj koji se dobija izbacivanjem svake druge cifre
    polaznog
3  broja. Cifre se posmatraju sa desna na levo.
    */
5
    #include <stdio.h>
7  #include <math.h>

9  int main() {

11     unsigned int x;
    int stepen_deset; // da li se radi o cifri jedinici, desetici,
    stotini itd...
13     int cifra;      // trenutna izdvojena cifra iz broja x
    int rbr; // redni broj cifre koju trenutno obradjujemo, gledano s
    desna na levo
15     unsigned int y; // broj dobijen nakon transformacije

17     /* Ucitavamo broj */
    printf("Unesite broj: ");
19     scanf("%d", &x);

21     if(x > 0) {
        /* Postavljamo vrednost stepena na 0 - to znaci da cemo prvo
        mnoziti sa
23         * 10^0=1 */
        stepen_deset = 0;

25         /* Postavljamo vrednost broja koji se formira na 0 */
        y = 0;
        /* Postavljamo redni broj pozicije na 0 */
27         rbr = 0;

31         /* Sve dok imamo cifara u zapisu broja */
        while(x > 0) {

33             /* Izdvajamo cifru */
            cifra = x%10;
35

```

```

37      /* Proveravamo da li je pozicija izdvojene cifre parna -
38         * cifre na parnim pozicijama zadržavamo
39         */
40      if(rbr % 2 == 0) {
41          /* I ako jeste */

42
43          /* Dodajemo izdvojenu cifru novom broju */
44          /* Neophodno je izvršiti "kastovanje" tipova, jer je double
45             povratni tip
46             * funkcije pow */
47          y += cifra * ((int) pow(10, stepen_deset));

48          /* Uvecavamo stepen zbog naredne cifre */
49          stepen_deset++;
50      }

51      /* Azuriramo redni broj cifre */
52      rbr++;
53      /* I pripremamo broj za naredno izdvajanje */
54      x /= 10;
55  }

56
57      /* Ispisujemo rezultat */
58      printf("%d\n", y);
59  }
60  else
61      printf("Nekorektan unos.\n");
62
63      return 0;
64  }

```

Rešenje 0.15

```

1  /* Sa standardnog ulaza unosi se neoznaceni ceo broj. Napisati program
2     koji formira i ispisuje broj koji se dobija
3     izbacivanjem cifara koje su jednake zbiru svojih suseda. Cifre se
4     posmatraju sa desna na levo. */
5
6  #include <stdio.h>
7
8  int main(){
9      unsigned n, novo_n;
10     int stepen;
11     int cifra_levo, cifra_sredina, cifra_desno;

12     /* Ucitavamo broj sa ulaza */
13     printf("Unesite broj: ");
14     scanf("%u", &n);
15

```

```

17  /* Stepen broja 10 sa kojim cemo mnoziti cifre izdvojenog broja */
    stepen=1;

19  /* Nova vrednost broja */
    novo_n=0;

21  /* Sve dok u zapisu broja imamo barem tri cifre */
23  while(n>99){
    /* Izdvajamo srednju cifru, cifru desno od nje i cifru levo od
       nje:
       npr. za trojku 583 8 je srednja cifra, 3 je cifra desno, a 5
       cifra levo */
25     cifra_desno=n%10;
27     cifra_sredina=(n/10)%10;
        cifra_levo=(n/100)%10;

29     /* U novi broj smestamo desnu cifru */
31     novo_n+=cifra_desno*stepen;

33     /* Azuriramo vrednost stepena */
        stepen=stepen*10;

35     /* Ako je srednja cifra jednaka zbiru leve i desne cifre */
37     if(cifra_levo+cifra_desno==cifra_sredina){

39         /* Treba izbaciti srednju cifru, pa broj n azuriramo tako sto
           ga podelimo sa 100 */
           n=n/100;
41     }
43     else{

45         /* Inace, zadržavamo srednju cifru i odbacujemo samo poslednju
           */
           n=n/10;
47     }

49     /* Na novi broj dodajemo preostali dvocifreni ili jednocifreni broj
       */
        novo_n=n*stepen+novo_n;

51     /* I ispisujemo rezultat */
53     printf("%d\n", novo_n);

55     return 0;

57 }

```

Rešenje 0.16

```

1  /* Napisati program koji proverava da li je dati prirodan broj
   palindrom. Broj
   je palindrom ako se isto cita i sa leve i sa desne strane. */
3
4  #include <stdio.h>
5  #include <math.h>
6
7  int main() {
8
9      int x;
10     int broj_cifara;
11     int min_stepen, max_stepen;
12     int pom;
13     int leva_cifra, desna_cifra;
14     int indikator;
15
16     printf("Unesite broj: ");
17     scanf("%d", &x);
18
19     /* Ako je korisnik uneo negativan broj, analiziramo njegovu
       apsolutnu
       * vrednost
       */
21    if(x < 0)
22        x=-x;
23
24
25
26     /* Odredjujemo broj cifara u zapisu broja x
       kako bismo mogli da izdvajamo istovremeno cifre i sa leve i sa
       desne
       strane
       */
29    broj_cifara = 0;
30    pom = x;
31    while(pom > 0) {
32        pom /= 10;
33        broj_cifara++;
34    }
35
36
37     /* Odredjujemo stepen koji stoji uz krajnju levu cifru broja */
38     max_stepen = (int) pow(10, broj_cifara-1);
39
40     /* Indikator je promenljiva koja ce nam ukazivati da li je broj
       * palindrom ili ne
       */
41
42     indikator=1;
43     while(x!=0 && indikator==1){
44         /* Izdvajamo levu cifru */
45         leva_cifra=x/max_stepen;
46         /* Izdvajamo desnu cifru */
47         desna_cifra=x%10;

```

```

49      /* Ako su cifre razlicite, odmah mozemo da zakljucimo da
      * broj nije palindrom i da prekinemo izvršavanje petlje */
51      if(leva_cifra!=desna_cifra){
          indikator=0;
53          break;
      }
55      /* Formiramo novu vrednost broja x tako sto odbacujemo
      * krajnju levu i krajnju desnu cifru */
57      x=(x%max_stepen-x%10)/10;
      /* I korigujemo maksimalan stepen tako dobijenog broja -
59      * delimo sa 100 jer smo odbacili 2 cifre */
      max_stepen=max_stepen/100;
61  }

63      /* Ispisujemo rezultat */
      if(indikator==1)
65          printf("Broj je palindrom!\n");
      else
67          printf("Broj nije palindrom!\n");

69      return 0;
71  }

```

Rešenje 0.17

```

1  /*
    Napisati program koji poziva korisnika da unese pozitivan ceo broj
    n,
3  a zatim za unetih n celih brojeva ispisuje sumu pozitivnih i sumu
    negativnih brojeva.
5
6  */
7
9  #include<stdio.h>
10
11  int main()
12  {
13      int n;
14      int x;
15      int suma_poz;
16      int suma_neg;
17      int i;
18
19      printf("Unesi pozitivan ceo broj:");
20      scanf("%d",&n);
21
22      suma_poz=0; /* promenljivim koje ce sadržati sumu se pre ulaska u
    petlju */
23      suma_neg=0; /* dodeljuje se 0 (neutral za sabiranje) */
    i=0;

```

```

25     while(i<n)
26     {
27         printf("Unesi ceo broj:");
28         scanf("%d", &x);
29
30         if (x<0)
31             suma_neg+=x;
32         else
33             suma_poz+=x;
34
35         i++;
36     }
37
38     printf(" Suma pozitivnih: %d\n Suma negativnih: %d\n",suma_poz,
39           suma_neg);
40     return 0;
41 }

```

Rešenje 0.18

```

/* Sa standardnog ulaza unosi se ceo pozitivan broj n, a potom i n
   celih
2  brojeva. Izracunati i ispisati zbir onih brojeva koji su neparni i
   negativni. */
4
#include <stdio.h>
6
int main(){
7     int n, i, x;
8     int zbir=0;
9
10    printf("Unesite broj n: ");
11    scanf("%d", &n);
12
13    printf("Unesite n brojeva: ");
14
15    /* Inicijalizujemo brojac kojim kontrolisemo broj učitavanja -
16       * treba da ih bude tacno n
17       */
18    i=0;
19    while(i<n){
20        /* Ucitavamo broj */
21        scanf("%d", &x);
22
23        /* Proveravamo da li broj negativan i neparan */
24        if(x<0 && x%2!=0){
25            /* Ako jeste, dodajemo ga na zbir */
26            zbir=zbir+x;
27        }
28    }

```

```

30     /* Uvecavamo brojac iteracija */
    i++;
32 }

34 /* Ispisujemo rezultat */
    printf("%d\n", zbir);

36 return 0;
}

```

Rešenje 0.19

```

1  /*
3  Napisati program koji omogućava korisniku da unosi cele brojeve dok
   ne unese nulu. Nakon toga ispisati proizvod onih unetih brojeva
   koji
   su pozitivni.
5  */

7  #include <stdio.h>
   int main()
9  {
   int x;
11  int p;

13  p=1;
   while (1) /* izraz 1 je konstantan; razlicit je od nule sto znaci
   da ga tumacimo kao tacnog */
15  {
   printf("Unesi jedan ceo broj:");
17   scanf("%d", &x);
   if (x==0) /* ukoliko je uneta nula */
19   break; /* break prekidamo petlju; izvršavanje se nastavlja
   od prve naredbe nakon petlje */

21   if (x<0) /* ukoliko je unet negativan broj, tu vrednost ne
   zelimo da pomnozimo sa ukupnim proizvodom p; zato moramo
   nastaviti dalje */
       continue; /* sa izvršavanjem petlje; continue prekida
   trenutnu iteraciju petlje tako sto preskace sve naredbe
23   koje nakon njega slede; izvršavanje se
   nastavlja od provere uslova petlje */
   p=p*x;
25  }

27  printf("Proizvod unetih brojeva je %d\n",p);

29  return 0;
}

```

Rešenje 0.20

```
2  /*
   Program izracunava minimum n unetih brojeva.
   Npr. za n=4 i brojeve 3 8 2 9 program ispisuje 2
   */
4  #include <stdio.h>
6  int main()
   {
8     int n, i;
     float x, min;

10

12     printf("Unesi n>0:");
     scanf("%d", &n);
14     if (n<=0)                                /* ako je unos neispravan */
     {
16         printf("Neispravan unos\n");
         return -1;                            /* prekidamo izvršavanje
   programa pomocu naredbe return */
18     }                                        /* u slucaju greske kao sto je
   neispravan unos vracamo vrednost -1 */
     printf("Unesi realan broj:");
20     scanf("%f", &x);                        /* prvi broj je unet izvan petlje */
     min=x;                                    /* kako bi bio njegova vrednost bila
   dodeljena promenljivoj min */
22                                     /* neophodno je da promenljiva min
   bude inicijalizovana pre ulaska u petlju */
                                     /* da bi uslov x<min mogao da bude
   ispitan u prvoj iteraciji */
24     i=0;
     while(i<(n-1))
     {
26         printf("Unesi realan broj:");
         scanf("%f", &x);
28         if(x<min)
             min=x;
30         i++;
32     }
     printf("Minimum je: %f\n", min);
34     return 0;
   }
```

Rešenje 0.22

```
1  /* Sa standardnog ulaza se unose realni brojevi sve do unosa broja 0.
   Napisati program koji izracunava i ispisuje
   aritmeticku sredinu unetih brojeva. */
3
   #include <stdio.h>
```



```

5  #include <math.h>

7  int main(){

9      float x;
      int broj_brojeva;
11     float suma;

13     /* Inicijalizujemo vrednosti */
      broj_brojeva=0;
15     suma=0;

17

      printf("Unesite brojeve: ");

19

      /* U petlji */
21     while(1){
          /* Ucitavamo broj sa ulaza */
23         scanf("%f", &x);

25         /* Ako je korisnik uneo 0, prekidamo sa petljom */
          if(x==0)
27             break;

29         /* Inace .. */

31         /* Procitani broj dodajemo na sumu */
          suma+=x;
33         /* I uvecavamo broj procitanih brojeva */
          broj_brojeva++;
35     }

37     /* Ispisujemo trazeni rezultat */
      printf("Aritmeticka sredina: %.4f\n", suma/broj_brojeva);
39

      return 0;
41 }

```

Rešenje 0.22

```

1  /* Sa standardnog ulaza se unose realni brojevi sve do unosa broja 0.
      Napisati program koji izracunava i ispisuje
      aritmeticku sredinu unetih brojeva. */
3

4  #include <stdio.h>
5  #include <math.h>

7  int main(){

9      float x;
      int broj_brojeva;

```

```

11  float suma;

13  /* Inicijalizujemo vrednosti */
   broj_brojeva=0;
15  suma=0;

17

19  printf("Unesite brojeve: ");

21  /* U petlji */
   while(1){
23     /* Ucitavamo broj sa ulaza */
       scanf("%f", &x);

25     /* Ako je korisnik uneo 0, prekidamo sa petljom */
       if(x==0)
27         break;

29     /* Inace .. */

31     /* Procitani broj dodajemo na sumu */
       suma+=x;
33     /* I uvecavamo broj procitanih brojeva */
       broj_brojeva++;
35  }

37  /* Ispisujemo trazeni rezultat */
   printf("Aritmeticka sredina: %.4f\n", suma/broj_brojeva);
39
   return 0;
41 }

```

Rešenje 0.23

```

1  /* Sa standardnog ulaza unosi se realan broj m, ceo pozitivan broj n
   i n realnih
   brojeva. Izracunati i ispisati koliko je brojeva medju unetima manje
   od zadatog
3  broja m. */

5  #include <stdio.h>

7  int main(){

9     float m, x;
       int n, i;
11     int broj_brojeva=0;

13     printf("Unesite broj m: ");
       scanf("%f", &m);
15

```

```

17     printf("Unesite broj n: ");
    scanf("%d", &n);

19     printf("Unesite n brojeva: ");
    /* Inicijalizujemo brojac kojim kontrolisemo broj ucitavanja -
21     * treba da ih bude tacno n
    */
23     i=0;
    while(i<n){
25         /* Ucitavamo broj */
        scanf("%f", &x);

27         /* Proveravamo da li je broj manji od zadatog broja m */
29         if(x<m){
            /* Ako jeste, uvecavamo brojac brojeva za 1 */
31             broj_brojeva++;
        }

33         /* Uvecavamo brojac iteracija */
35         i++;
    }

37     /* Ispisujemo rezultat */
39     printf("%d\n", broj_brojeva);

41     return 0;
}

```

Rešenje 0.62

Rešenje 0.62

Rešenje 0.26

```

/* Sa standardnog ulaza se unosi ceo pozitivan broj n, a zatim i n
   celih brojeva. Napisati program koji ispisuje
2 broj sa najvecom cifrom desetica. Ukoliko ima vise takvih, ispisati
   prvi. */

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

6
8 int main(){
10     int n;
    int x, x_desetica;
    int max_desetica, broj;
12     int i;

```

```

14  /* Citamo vrednost sa ulaza */
    printf("Unesite broj n: ");
16  scanf("%d", &n);

18  /* Postavljamo maksimalnu cifru desetice na 0 - 0 je svakako
    najmanja cifra pa je pocetna vrednost neutralna tj.
    ne moze da utice na maksimum koji izracunavamo. Nije uvek zgodno
    pretpostaviti da je maksimalna vrednost 0. Na primer,
20  ako trazimo maksimum celih brojeva, a korisnik unese -32 -7 i -22,
    maksimalni je broj -7 */
    max_desetica=0;

22

24  /* Ucitavamo broj po broj */
    printf("Unesite n brojeva: ");
    for(i=0; i<n; i++){
26        scanf("%d", &x);

28        /* Izdvajamo cifru desetica procitanog broja */
        x_desetica=(abs(x)/10)%10;

30        /* Ako je ona veca od maksimalne cifre desetica */
        if(x_desetica>max_desetica){
32            /* Cuvamo je */
            max_desetica=x_desetica;
34            /* Ali zbog ispisa, cuvamo i broj u kojem se ona pojavljuje */
            broj=x;
36        }
38    }

40    /* Ispisujemo rezultat */
    printf("%d\n", broj);

42    return 0;

44 }

```

Rešenje 0.27

```

/* Sa standardnog ulaza se unosi ceo pozitivan broj n, a zatim i n
celih brojeva. Napisati program koji ispisuje
2 broj sa najvećim brojem cifara. Ukoliko ima više takvih, ispisati
prvi. */

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

6
8 int main(){

10     int n;
    int x, x_kopija, br_cifara;
    int max_br_cifara, broj;

```

```

12  int i;

14  /* Citamo vrednost sa ulaza */
printf("Unesite broj n: ");
16  scanf("%d", &n);

18  /* Postavljamo maksimalan broj cifara na 0 - svaki broj ima vise
    od 0 cifara pa je ova vrednost neutralna */
max_br_cifara=0;

20  /* Ucitavamo broj po broj */
printf("Unesite n brojeva: ");
22  for(i=0; i<n; i++){
24      scanf("%d", &x);

26      /* Odredjujemo broj cifara unetog broja x */
x_kopija=abs(x);
28      br_cifara=0;
while(x_kopija!=0){
30          x_kopija=x_kopija/10;
          br_cifara++;
32      }
/* Ako je broj cifara unetog broja veci od maksimalnog */
34  if(br_cifara>max_br_cifara){
/* Cuvamo ga */
max_br_cifara=br_cifara;
/* I zbog ispisa rezultata, cuvamo i originalni broj */
38  /* Zbog ovoga smo morali i da racunamo broj cifara nad kopijom
    broja x kako ne bismo promenili njegovu vrednost */
    broj=x;
40  }
}

42  /* Ispisujemo rezultat */
printf("%d\n", broj);

44  return 0;

46  }
48  }

```

Rešenje 0.28

```

/* Sa standardnog ulaza se unosi ceo pozitivan broj n, a zatim i n
celih brojeva. Napisati program koji ispisuje
2 broj sa najvecom vodecom cifrom. Vodeca cifra je prva cifra iz zapisa
    broja. Ukoliko ima vise takvih, ispisati
    prvi. */

4
#include <stdio.h>
6 #include <math.h>

```

```

8  int main(){
10     int n;
11     int x, x_kopija;
12     int broj;
13     int vodeca_cifra, max_vodeca_cifra;
14     int i;

16     /* Citamo vrednost sa ulaza */
17     printf("Unesite broj n: ");
18     scanf("%d", &n);

20     /* Postavljamo maksimalnu vodecu cifru na 0 - cifre broja su vece
        ili jednake od 0 pa je ova vrednost neutralna */
21     max_vodeca_cifra=0;

22     /* Ucitavamo broj po broj */
23     printf("Unesite n brojeva: ");
24     for(i=0; i<n; i++){
25         scanf("%d", &x);

28         /* Odredjujemo vodecu cifru broja */
29         x_kopija=abs(x);
30         while(x_kopija>10){
31             x_kopija=x_kopija/10;
32         }
33         vodeca_cifra=x_kopija;

34         /* Ako je izdvojena cifra veca od maksimalne vodece cifre */
35         if(vodeca_cifra>max_vodeca_cifra){
36             /* Cuvamo je */
37             max_vodeca_cifra=vodeca_cifra;
38             /* I zbog ispisa, cuvamo i broj u kojem se ona pojavljuje */
39             /* Zbog ovoga smo morali i da racunamo vodecu cifru nad kopijom
                broja x kako ne bismo promenili njegovu vrednost */
40             broj=x;
41         }
42     }

44     /* Ispisujemo rezultat */
45     printf("%d\n", broj);

48     return 0;

50 }

```

Rešenje 0.29

```

/* Sa standardnog ulaza se unose celi pozitivni brojevi n (n > 1) i d
   , a zatim i n celih brojeva. Napisati program

```

```

2  koji izracunava koliko ima parova uzastopnih brojeva medju unetim
   brojevima koji se nalaze na rastojanju d.
   Rastojanje izmedu brojeva je definisano sa  $d(x, y) = |y - x|$ . Rezultat
   ispisati na standardni izlaz. */
4
6  #include <stdio.h>
   #include <math.h>
8  int main(){
10     int n;
12     int d;
14     int x, y;
16     int broj_parova;
18     int i;

20     /* Ucitavamo vrednosti sa ulaza */
22     printf("Unesite brojeve n i d: ");
24     scanf("%d %d", &n, &d);

26     /* Inicijalizujemo broj parova */
28     broj_parova=0;

30     printf("Unesite n brojeva: ");

32     /* Ucitavamo prvi broj */
34     scanf("%d", &x);

36     for(i=1; i<n; i++){
38         /* Ucitavamo naredni broj */
40         scanf("%d", &y);

42         /* Ako su brojevi na rastojanju d */
44         if(abs(y-x)==d)
46             /* Treba uvecati broj parova */
48             broj_parova++;

49         /* Cuvamo broj iz tekuce iteracije kako bismo mogli da ga
50            upotrebimo u narednoj iteraciji */
51         x=y;
52     }

53     /* Ispisujemo rezultat */
54     printf("Broj parova: %d\n", broj_parova);

55     return 0;
56 }

```

Rešenje 0.30

```

1  /* Sa standardnog ulaza se unose celi brojevi sve do unosa broja 0.
   Napisati program koji izracunava i ispisuje
   razliku najveceg i najmanjeg unetog broja. */
3
4  #include <stdio.h>
5  #include <math.h>
6
7  int main(){
8
9      int x;
10     int min, max;
11
12     printf("Unesite brojeve: ");
13
14     /* Prvi broj učitavamo izvan petlje zbog inicijalizacije maksimuma
       i minimuma */
15     scanf("%d", &x);
16     max=x;
17     min=x;
18
19     /* U petlji smo sve dok ne procitamo broj 0 */
20     while(x!=0){
21
22         /* Proveravamo da li je procitani broj veci od aktuelnog
           maksimuma */
23         if(x>max)
24             max=x;
25         /* Proveravamo da li je procitani broj manji od aktuelnog
           minimuma */
26         if(x<min)
27             min=x;
28
29         /* Učitavamo naredni broj */
30         scanf("%d", &x);
31     }
32
33     /* Ispisujemo razliku najveceg i najmanjeg broja */
34     printf("Razlika: %d\n", max-min);
35
36     return 0;
37 }

```

Rešenje 0.31

```

1  /*
   Napisati program koji omogućava korisniku da unosi karaktere dok ne
   zada tacku i ukoliko je karakter malo slovo,
3  ispisuje odgovarajuće veliko, ukoliko je karakter veliko slovo
   ispisuje odgovarajuće malo, a u suprotnom ispisuje
   isti karakter kao i uneti.

```



```

5  */
7  #include <stdio.h>
9  int main()
10 {
11     int c;
13     /* funkcija getchar učitava jedan karakter.
14        naredbom dodele (c=getchar()) promenljivoj c bice dodeljena
15        vrednost
16        ascii koda unetog karaktera
17        obratiti paznju na zagrade!
18    */
19    while((c=getchar())!='.')
20    {
21        if (c>='A' && c<='Z')
22            putchar(c+'a'-'A'); /* Razlika izmedju ascii koda svakog malog
23                                i odgovarajuceg velikog slova
24                                je konstanta koja se moze sracunati
25                                izrazom 'a'-'A' (i iznosi 32) */
26        else if (c>='a' && c<='z')
27            putchar(c-'a'+'A');
28        else
29            putchar(c);
30    }
31    return 0;
32 }

```

Rešenje 0.32

```

1  /*
2     Napisati program koji omogućava korisniku da unosi karaktere dok
3     ne zada EOF a potom ispisuje broj velikih slova, broj malih slova
4     ,
5     broj cifara, broj belina i zbir cifara.
6  */
7
8  #include <stdio.h>
9
10 int main()
11 {
12     /* promenljivoj c dodelicemo povratnu vrednost funkcije getchar()
13        funkcija getchar() učitava jedan karakter sa standardnog ulaza
14        i vraca njegov ascii kod; povratna vrednost funkcije getchar je
15        int, pa i promenljiva c mora biti tipa int
16    */
17
18     int c;
19
20     /* brojac moraju biti inicijalizovani na 0 */

```

```

20 int br_v=0;
22 int br_m=0;
24 int br_c=0;
26 int br_b=0;
28 int br_k=0;
30 int suma=0;

26 while((c=getchar())!=EOF) /* petlja se završava kada
korisnik ne unese karakter, već zada konstantu EOF */
{
    /* ova konstanta se zadaje
    kombinacijom tastera CTRL+D. U tom slučaju, getchar() vraća -1*/
28 if (c>='A' && c<='Z')
    br_v++; /* <=> br_v = br_v+1; */
30 else if (c>='a' && c<='z')
    br_m++;
32 else if (c>='0' && c<='9')
    {
34 br_c++;
suma=suma+c-'0'; /* funkcija getchar() vraća ascii
kod unetog karaktera; ascii kodovi cifara 0,1,...,9
36 su redom 48,49,...,57; Na primer,
za unetu 1
promenljiva c će imati vrednost
49. Zbog toga bi bilo pogrešno računati
38 zbir kao zbir=zbir+c. Promenljivu zbir zato
računamo kao zbir=zbir+(c-'0')
jer c-'0' će za unetu 0 proizvesti 48-'0' što je
0,
40 za unetu 1 49-'0' što je 1, za unetu 2 50-'0' što
je 2, ...*/
    }
42 else if (c=='\t' || c=='\n' || c==' ')
    br_b++;
44 br_k++;
46 }

48 printf("velika: %d, mala: %d, cifre: %d, beline: %d, svi: %d\n",
br_v, br_m, br_c, br_b, br_k);
printf("suma cifara: %d\n", suma);
50 return 0;
52 }

```

Rešenje 0.33

Rešenje 0.34

```

1 /* Sa standardnog ulaza se unosi ceo broj n, a zatim i n karaktera.
   Napisati program koji proverava da li se od

```

```

3  unetih karaktera moze napisati rec Zima. */
5  #include <stdio.h>
5  #include <math.h>
7  int main(){
9      int n;
9      int broj_Z, broj_i, broj_m, broj_a;
11     char novi_red, c;
11     int i;
13
13     broj_Z=0;
15     broj_i=0;
15     broj_m=0;
17     broj_a=0;
19
19     /* Ucitavamo broj karaktera */
21     printf("Unesite broj: ");
21     scanf("%d", &n);
23
23     /* Ucitavamo karakter po karakter */
25     for(i=0; i<n; i++){
25         printf("Unestite %d. karakter: ", i+1);
27         /*
27          Prvo citamo znak za novi red koji je ostao neprocitan nakon
29          pritiska Enter tastera
29          posle prethodnog unosa, pa tek onda citamo karakter koji treba
29          obradivati
29          */
31         scanf("%c%c", &novi_red, &c);
33
33         /* Analiziramo karakter */
35         switch(c){
35             case 'Z':
37                 broj_Z++;
37                 break;
39             case 'i':
39                 broj_i++;
39                 break;
41             case 'm':
41                 broj_m++;
43                 break;
43             case 'a':
45                 broj_a++;
45                 break;
47         }
47     }
49
49     /* Ako imamo barem jedno veliko slovo z i barem po jedno malo slovo
49        i, m i a */

```

```

51     if(broj_Z && broj_i && broj_m && broj_a){
        /* Zaključujemo da se rec može napisati */
53     printf("Može se napisati rec Zima.\n");
    }
55     else{
        /* Inace, obavestavamo korisnika da je to nemoguće */
57     printf("Ne može se napisati rec Zima.\n");
    }
59
    return 0;
61 }

```

Rešenje 0.35

```

/*
2   Napisati program koji za uneti pozitivan ceo broj
   izracunava njegov faktorijel. Testirati program
4   za razlicite vrednosti promenljive x. Obratiti paznju
   da pocev od 23! dolazi do prekoracenja.
6 */

8 #include<stdio.h>

10 int main()
   {
12     int x;
        unsigned long f;
14     int i;
        int original;

16     printf("Unesi x>=0:");
18     scanf("%d",&x);

20     original=x;
        f=1;
22     if (x<0)
        printf("Nekorektan unos\n");
24     else
        {
26         while (x>1)
            {
28             f=f*x; /* vrednost izraza sa desne strane naredbe dodele
                        dodeljujemo promenljivoj sa leve strane naredbe
dodele
30                 */
                x--; /* operator -- umanjuje vrednost promenljive x za 1
32                 naredba x--; ima isti efekat kao x-=1;
                        ili x=x-1;
34                 */
            }
        }
   }

```

```

36     printf("%d! = %lu\n",x,f);          /* nekorektno: vrednost
promenljive x je unistena */
    printf("%d! = %lu\n",original,f); /* korektno: promenljiva
original sadrzi vrednost promenljive x pre ulaska u petlju */
38
40     }
42     return 0;
}

```

Rešenje 0.36

```

1  /* Sa standradnog ulaza unose se realan broj x i ceo neoznaceni broj n
   . Napisati
   program koji izracunava x^n */
3
4  #include <stdio.h>
5
6  int main(){
7
8      int n;
9      float x;
10     float vrednost;
11     unsigned exp;
12
13     /* Ucitavaju se brojevi x i n */
14     printf("Unesite redom brojeve x i n: ");
15     scanf("%f %d", &x, &n );
16
17     /* Pocetna vrednost stepena koji se racuna */
18     vrednost=1;
19
20     for(exp=1; exp<=n; exp++)
21         vrednost=vrednost*x;
22
23     /* Stampamo rezultat */
24     printf("%f\n",vrednost);
25
26     return 0;
27 }

```

Rešenje 0.37

```

1  /* Sa standradnog ulaza unose se realan broj x i ceo broj n. Napisati
   program koji izracunava x^n */
3
4  #include <stdio.h>
5

```

```

7  int main(void){
9      int n, n_abs;
10     float x;
11     float vrednost;
12     unsigned exp;

13     /* Ucitavaju se brojevi x i n */
14     printf("Unesite redom brojeve x i n: ");
15     scanf("%f %d", &x, &n );

17     /* Pocetna vrednost stepena koji se racuna */
18     vrednost=1;

19     /* Stepenovanje */
20     n_abs=abs(n);
21     for(exp=1; exp<=n_abs; exp++)
22         vrednost=vrednost*x;

23     /* Stampamo rezultat */
24     if(n<0){
25         printf("%.3f\n",1/vrednost);
26     }
27     else{
28         printf("%.3f\n",vrednost);
29     }

30     return 0;
31 }

```

Rešenje 0.39

```

/*
2  a) Napisati program za uneti pozitivan ceo broj n ispisuje zbir
3     s = 1+2^3+3^3+...+n^3. Na primer, za n=4, izlaz iz programa
4     treba da bude:
5     Suma kubova od 1 do 4 je 100
6  b) Modifikovati program tako da ispisuje zbir s = 1+2^3+3^3+...+k
7     ^3
8     za svako i od 1 do n. Na primer, za n=4, izlaz iz programa
9     treba da
10 bude:
11 i=1, n=1
12 i=2, n=9
13 i=3, n=36
14 i=4, n=100
15
16 */
17 #include <stdio.h>

```

```

18 int main()
19 {
20     int n;
21     int i;
22     int s;
23
24     printf("Unesite jedan pozitivan ceo broj:");
25     scanf("%d", &n);
26
27     if (n<0)
28         return -1;
29
30     i=1;
31     s=0; /* inicijalizacija promenljive u kojoj se cuva suma kubova */
32
33     for(i=1;i<=n;i++)
34     {
35         s+=i*i*i;
36         /* b) */
37         printf("i=%d, s=%d\n", i, s);
38     }
39     /* a) */
40     printf("Suma kubova od 1 do %d: %d\n", n, s);
41     return 0;
42 }

```

Rešenje 0.39

```

1  /*
2      a) Napisati program za uneti pozitivan ceo broj n ispisuje zbir
3          s = 1+2^3+3^3+...+n^3. Na primer, za n=4, izlaz iz programa
4          treba da bude:
5          Suma kubova od 1 do 4 je 100
6      b) Modifikovati program tako da ispisuje zbir s = 1+2^3+3^3+...+k
7          ^3
8          za svako i od 1 do n. Na primer, za n=4, izlaz iz programa
9          treba da
10         bude:
11         i=1, n=1
12         i=2, n=9
13         i=3, n=36
14         i=4, n=100
15
16     */
17
18     #include <stdio.h>
19
20     int main()
21     {
22         int n;

```

```

21  int i;
    int s;

23

25  printf("Unesite jedan pozitivan ceo broj:");
    scanf("%d", &n);

27

    if (n<0)
29         return -1;

31  i=1;
    s=0; /* inicijalizacija promenljive u kojoj se cuva suma kubova */

33

    for(i=1;i<=n;i++)
35    {
        s+=i*i*i;
        /* b) */
        printf("i=%d, s=%d\n", i, s);
37    }
    /* a) */
41    printf("Suma kubova od 1 do %d: %d\n", n, s);
    return 0;
43 }

```

Rešenje 0.40

```

1  /* Sa standardnog ulaza unose se realan broj x i ceo neoznacen broj n
   . Napisati
   program koji izracunava sumu  $S=x+2*x^2+3*x^3+...+n*x^n$  */

3

    #include <stdio.h>

5

    int main(){
6
7        unsigned n, i;
        float x, S, stepen;

9

        printf("Unesite redom brojeve x i n: ");
11       scanf("%f %u", &x, &n);

13

        /* Inicijalizujemo sumu koju racunamo */
        S=0;

15

        /* Stepen promenljiva ce sadrzati vrednosti stepena  $x^n$  -
        * pocetna vrednost joj je 1 */
        stepen=1;

17

19       for(i=1; i<=n; i++){
21           stepen=stepen*x;
           S=S+i*stepen;
23       }

```



```

25     printf("S=%f\n", S);
27     return 0;
}

```

Rešenje 0.41

```

1  /* Sa standardnog ulaza unose se realan broj x i ceo neoznacen broj n
   *
   * Napisati program koji izracunava sumu  $S=1+1/x+1/x^2+1/x^3+\dots+1/x^n$ 
   */
3
4  #include <stdio.h>
5  int main(){
6      unsigned n, i;
7      float x, S, stepen;
8
9      printf("Unesite redom brojeve x i n: ");
10     scanf("%f %u", &x, &n);
11
12     S=1;
13     stepen=1;
14     for(i=1; i<=n; i++){
15         stepen=stepen*x;
16         S=S+1/stepen;
17     }
18
19     printf("S=%f\n", S);
20
21     return 0;
22 }

```

Rešenje 0.42

```

1  /* Napisati program koji sa zadatom tacnoscu izracunava sumu
2   $S=1+x+x^2/2!+x^3/3!+\dots+x^n/n! + \dots$ 
3  /* Napomena: ovo je razvoj funkcije  $e^x$  */
4
5  #include <stdio.h>
6  #include <math.h>
7  int main(){
8      int n, i, faktoriyel;
9      float S;
10     float x, eps, stepen;
11
12     printf("Unesite x: ");
13     scanf("%f", &x);
14
15     printf("Unesite tacnost eps: ");

```

```

16     scanf("%f", &eps);

18

19     /* Tacnost izracunavanja je zadovoljena ako je apsolutna vrednost
20      * razlika suma
21      * u dvema uzastopnim iteracijama manja od zadate tacnosti;
22      * Odavde se izvodi da apsolutna vrednost opsteg clana sume
23      * mora da bude manja od zadate tacnosti da bi uslov bio ispunjen
24      */

25
26     S=1;
27     faktorijel=1;
28     stepen=x;
29     i=2;
30     while(fabs(stepen/faktorijel)>eps){
31         S=S+stepen/faktorijel;
32         stepen=stepen*x;
33         faktorijel=faktorijel*i;
34         i++;
35     }

36     printf("S=%f\n", S);

38     return 0;

40 }

```

Rešenje 0.43

```

/* Napisati program koji sa zadatom tacnosu izracunava sumu
2  S=1-x+x^2/2!-x^3/3!+... */
/* razvoj funkcije sin(x) */

4

5  #include <stdio.h>
6  #include <math.h>
7  int main(){
8      int n, i, faktorijel, znak;
9      float S;
10     float x, eps, stepen;

11
12     printf("Unesite x: ");
13     scanf("%f", &x);

14
15     printf("Unesite tacnost eps: ");
16     scanf("%f", &eps);

17
18     /* Tacnost izracunavanja je zadovoljena ako je apsolutna vrednost
19      * razlika suma
20      * u dvema uzastopnim iteracijama manja od zadate tacnosti;
21      * Odavde se izvodi da apsolutna vrednost opsteg clana sume
22      * mora da bude manja od zadate tacnosti da bi uslov bio ispunjen
23      */

```

```

24     S=1;
26     faktorijel=1;
    stepen=x;
28     i=2;
    znak=-1;
30     while(fabs(stepen/faktorijel)>eps){
        S=S+znak*stepen/faktorijel;
32         stepen=stepen*x;
        faktorijel=faktorijel*i;
34         znak=-znak;
        i++;
36     }

38     printf("S=%f\n", S);

40     return 0;
}

```

Rešenje [0.62](#)

Rešenje [0.62](#)

Rešenje [0.62](#)

Rešenje [0.62](#)

Rešenje [0.62](#)

Rešenje [0.62](#)

Rešenje [0.62](#)

Rešenje [0.52](#)

```

1  #include <stdio.h>

3  int main(){

5      int n, i, j;

7      printf("Unesite broj n: ");
      scanf("%d", &n);

9

```

```

11  /* Krstice koje iscrtavamo mozemo posmatrati kao dijagonale
12     kvadrata dimenzije n */
13
14  /* Prolazimo kroz sve vrste kvadrata */
15  for(i=1; i<=n; i++){
16
17      /* Prolazimo kroz sve kolone kvadrata */
18      for(j=1; j<=n; j++){
19
20          /* Ako se nalazimo na glavnoj ili sporednoj dijagonali */
21          if(i==j || i+j==n+1)
22              /* Stampamo zvezdu */
23              putchar('*');
24          else
25              /* Inace, stampamo blanko znak */
26              putchar(' ');
27      }
28
29      /* Nakon uspesno iscrtane vrste, stampamo znak za novi red */
30      putchar('\n');
31  }
32
33  return 0;
34  }

```

Rešenje 0.52

```

1 #include <stdio.h>

3 int main(){

5     int n, i, j;

7     printf("Unesite broj n: ");
8     scanf("%d", &n);

10    /* Krstice koje iscrtavamo mozemo posmatrati kao dijagonale
11       kvadrata dimenzije n */

13    /* Prolazimo kroz sve vrste kvadrata */
14    for(i=1; i<=n; i++){

16        /* Prolazimo kroz sve kolone kvadrata */
17        for(j=1; j<=n; j++){

19            /* Ako se nalazimo na glavnoj ili sporednoj dijagonali */
20            if(i==j || i+j==n+1)
21                /* Stampamo zvezdu */
22                putchar('*');
23            else
24                /* Inace, stampamo blanko znak */

```

```
        putchar(' ');
25     }

27     /* Nakon uspesno iscrtane vrste, stampamo znak za novi red */
    putchar('\n');
29 }

31 return 0;
}
```

Rešenje [0.62](#)

Rešenje [0.62](#)

Rešenje [0.55](#)

Rešenje [0.62](#)

Rešenje [0.62](#)

Rešenje [0.62](#)

Rešenje [0.62](#)

Rešenje [0.62](#)

Rešenje [0.62](#)

Rešenje [0.62](#)