

PROGRAMIRANJE 1

**Milena Vujošević Jančić, 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

1	Ulaz i izlaz programa	vii
1.1	Datoteke	vii
1.1.1	Unos karakter po karakter	vii
1.1.2	Rec po rec	xii
1.1.3	Broj po broj	xv
1.1.4	Strukture	xvii
1.1.5	Linija po linija	xxii
1.2	Rešenja	xxvi

1

Ulaz i izlaz programa

1.1 Datoteke

Jovana: poruke o gresci:

- ako je dat fiksni naziv datoteke, da li u test primere stavljamo slucaj kada datoteka ne postoji?
- da li je poruka o gresci uvek -1 ili da navodimo customized?
- da li naglasavati da je tekstualna datoteka (u zadacima sa i smerom nekad stoji na pocetku zadatka 'tekstualna datoteka', kod nas ne)
- kada treba da se ispise minimum ili maksimum nekakvog niza, ako nije precizirano sta se radi ako vise elemenata ima maksimalnu/minimalnu vrednost i nema resenja, da li da preciziramo u zadatku?)

1.1.1 Unos karakter po karakter

Zadatak 1.1.1 Napisati program koji prepisuje sadržaj datoteke *ulaz.txt* u datoteku *izlaz.txt* karakter po karakter.

Jovana: v301, karakter po karakter, prepisivanje, fiksni naziv

Primer 1

```
ULAZ.TXT
Danas je 21. mart.
To je prvi dan proleca.
IZLAZ.TXT
Danas je 21. mart.
To je prvi dan proleca.
```

Primer 2

```
ULAZ.TXT
Ispit iz Programiranja 1 je zakazan za 10. jun.
IZLAZ.TXT
Ispit iz Programiranja 1 je zakazan za 10. jun.
```

Zadatak 1.1.2 Napisati program koji prepisuje svaki treći karakter datoteke *ulaz.txt* u datoteku *izlaz.txt*.

Jovana: p302, karakter po karakter, prepisivanje, fiksni naziv

Primer 1	Primer 2	Primer 3
<pre>ULAZ.TXT Volim programiranje. IZLAZ.TXT Vipgmae</pre>	<pre>ULAZ.TXT abcdefghi 123456789 IZLAZ.TXT adg 147</pre>	<pre>ULAZ.TXT U Beogradu ce biti suncan i lep dan. IZLAZ.TXT Ueruei nn pa</pre>

Zadatak 1.1.3 Napisati program koji šifrira sadržaj datoteke *plain.txt* tako što svako slovo ciklično zamenjuje njegovim prethodnikom suprotne veličine i upisuje u datoteku *sifra.txt*. Na primer, 'b' se zamenjuje sa 'A', 'B' sa 'a', 'a' sa 'Z', 'A' sa 'z', itd. Ostali karakteri ostaju nepromenjeni. Podrazumevati da se na sistemu koristi tabela karaktera ASCII. U slučaju da datoteka *plain.txt* ne postoji, napisati na standardni izlaz poruku o grešci.

Jovana: p3id17, karakter po karakter, prepisivanje, fiksno ime, nema resenje

Jovana: da li moramo da naglasavamo ovo za ASCII?

Primer 1	Primer 2	Primer 3
<pre>PLAIN.TXT ABC.123.xyz SIFRA.TXT zab.123.WXY</pre>	<pre>PLAIN.TXT a=x+y; x=b+5; SIFRA.TXT Z=W+X; W=A+5;</pre>	<pre>PLAIN.TXT NE POSTOJI IZLAZ: Ulazna datoteka ne postoji</pre>

Zadatak 1.1.4 Sa standarnog ulaza učitavaju se imena dve tekstualne datoteke i jedan karakter. Napisati program koji prepisuje sadržaj prve datoteke u drugu na sledeći način:

- ukoliko je učitani karakter u, sva mala slova zamenjuje velikim
- ukoliko je učitani karakter l, sva velika slova zamenjuje malim

U slučaju greške ispisati -1. Greška može biti neuspešno otvaranje datoteke ili pogrešno zadat karakter. Maksimalna dužina naziva datoteka je 20 karaktera.

Jovana: p3iv3, karakter po karakter, prepisivanje, ima resenje

Jovana: da li da izbrisemo sta moze biti greska?

Primer 1

```
INTERAKCIJA SA PROGRAMOM:
  ulaz.txt izlaz.txt u
ULAZ.TXT
  danas je lep dan
  i Ja zelim
  da postanem programer
IZLAZ.TXT
  DANAS JE LEP DAN
  I JA ZELIM
  DA POSTANEM PROGRAMER
```

Primer 2

```
INTERAKCIJA SA PROGRAMOM:
  prva.dat druga.dat l
PRVA.DAT
  Cena soka je 30
  Cena vina je 150
  Cena limunade je 200
  Cena sendvica je 120
DRUGA.DAT
  cena soka je 30
  cena vina je 150
  cena limunade je 200
  cena sendvica je 120
```

Primer 3

```
INTERAKCIJA SA PROGRAMOM:
  primer.c prazna.txt V
PRIMER.C
  #include <stdio.h>
  int main()
  {
  }
PRAZNA.TXT

IZLAZ:
-1
```

Zadatak 1.1.5 Napisati program koji za dve datoteke čija se imena unose sa standardnog ulaza, radi sledeće:

- za svaku cifru u prvoj datoteci, u drugu datoteku upisuje 0
- za svako slovo u prvoj datoteci, u drugu datoteku upisuje 1
- za sve ostale karaktere u prvoj datoteci, u drugu datoteku upisuje 2

Maksimalna dužina naziva datoteka je 20 karaktera. U slučaju da prva datoteka ne postoji, napisati na standardni izlaz poruku o grešci.

Jovana: p3id16, karakter po karakter, prepisivanje, nema resenje

Jovana: Dodat je deo zadatka za slucaj da datoteka ne postoji.

Jovana: ispraviti ovoliki prored izmedju stavki u okviru itemize (to se menja u zavisnosti od prostora na stranici)

Primer 1

```
INTERAKCIJA SA PROGRAMOM:
  prva.dat druga.dat
PRVA.DAT
  abc.123.[]
  567.ABC.
DRUGA.DAT
  111200022220002111222
```

Primer 2

```
INTERAKCIJA SA PROGRAMOM:
  ulaz.txt izlaz.txt
ULAZ.TXT
  18. februar 2019.
IZLAZ.TXT
  11220000000211112
```

Primer 3

```
INTERAKCIJA SA PROGRAMOM:
  in.txt out.txt
IN.TXT NE POSTOJI
IZLAZ:
  Ulazna datoteka ne postoji
```

Zadatak 1.1.6 Napisati program koji prebrojava mala slova u datoteci *test.txt*.

Jovana: p301, karakter po karakter, brojanje, fiksni naziv

Primer 1

```
TEST.TXT
  Abcd EFGH+ijKLMN

IZLAZ:
  Broj malih slova je: 5
```

Primer 2

```
TEST.TXT
  PrograMiranje

IZLAZ:
  Broj malih slova je: 11
```

Primer 3

```
TEST.TXT
  123456
  ABCDEF

IZLAZ:
  Broj malih slova je: 0
```

Zadatak 1.1.7 Napisati program koji u datoteci čije se ime unosi sa standardnog ulaza prebrojava koliko se puta svaka cifra pojavljuje i na standardni izlaz ispisuje cifru sa najvećim brojem pojavljivanja. Ukoliko ima više takvih cifara, ispisati sve. U slučaju da nema cifara u datoteci ili u slučaju greške pri otvaranju, ispisati -1 na standardni izlaz. Maksimalna dužina naziva datoteka je 20 karaktera.

Jovana: p2id14, karakter po karakter, brojanje, nema resenje (domaci)

Jovana: opet, da obrisemo o kojoj se gresci radi, da stavimo samo u slucaju gresaka?

Primer 1

```
INTERAKCIJA SA PROGRAMOM:
  ulaz.txt
ULAZ.TXT
  danas je lep dan
  i ja zelim
  da postanem programer
IZLAZ:
  -1
```

Primer 2

```
INTERAKCIJA SA PROGRAMOM:
  prva.dat
PRVA.DAT
  Cena soka je 30
  Cena vina je 150
  Cena limunade je 200
  Cena sendvica je 120
IZLAZ:
  0
```

Primer 3

```
INTERAKCIJA SA PROGRAMOM:
  primer.c
PRIMER.C
  1 22 333.444
IZLAZ:
  3 4
```

Zadatak 1.1.8 Napisati program koji u datoteci čije je ime dato kao argument komandne linije proverava da li su zagrade pravilno uparene. Ukoliko nisu zadati svi argumenti komandne linije, ispisati poruku o grešci na standardni izlaz.

Jovana: p3x2, argumenti, karakter po karakter, brojanje, nema resenja

Primer 1

```
POKRETANJE: ./a.out zagrade.txt
ZAGRADE.TXT
  ab( cd) ..
  ((3+4)*5+1)*9
IZLAZ:
  jesu
```

Primer 2

```
POKRETANJE: ./a.out primer2.dat
PRIMER2.DAT
  (7+8
  nisu(
  uparene
IZLAZ:
  nisu
```

Primer 3

```
POKRETANJE: ./a.out primer3.dat
PRIMER3.DAT
)) 7 + 6 ((
IZLAZ:
nisu
```

Primer 4

```
POKRETANJE: ./a.out
IZLAZ:
greska
```

Zadatak 1.1.9 Potrebno je napisati program koji prebrojava slova i cifre u datoteci.

- Napisati C funkciju `int unesiSkup(char *s, FILE* f)` kojom se unosi skup elemenata iz datoteke `F`. Skup se predstavlja kao niz karaktera, pri čemu su dozvoljeni elementi skupa mala i velika slova abecede, kao i cifre. Unos se prekida kada se naiđe na znak za novi red ili nedozvoljeni karakter za skup (maksimalan broj elemenata skupa je 1000). Funkcija vraća broj elemenata skupa koji su uspešno učitani.
- Napisati funkciju `void prebroj(char *s, int *br_slova, int *br_cifara)` kojom se određuje broj slovnih elemenata skupa (velikih ili malih slova) kao i broj cifara u skupu.
- Napisati glavni program koji koristeći prethodne funkcije prebrojava cifre i slova u datoteci čije se ime unosi kao argument komandne linije i ispisuje dobijene vrednosti na standardni izlaz. U slučaju greške, ispisati -1 na standardni izlaz.

Jovana: x4, karakter, brojanje, argumenti, nema resenje

Primer 1

```
POKRETANJE: ./a.out skup.txt
SKUP.TXT
abc56ighj9012hjFGHH
IZLAZ:
broj slova: 13
broj cifara: 6
```

Primer 2

```
POKRETANJE: ./a.out skup2.txt
SKUP2.TXT
ovdeimamo$dolar
IZLAZ:
broj slova: 9
broj cifara: 0
```

Primer 3

```
POKRETANJE: ./a.out skup3.txt
SKUP3.TXT
broj3
broj5
IZLAZ:
broj slova: 4
broj cifara: 1
```

Primer 4

```
POKRETANJE: ./a.out skup4.txt
SKUP4.TXT
11.2.2019.
IZLAZ:
broj slova: 0
broj cifara: 2
```

Primer 5

```
POKRETANJE: ./a.out skup5.txt
SKUP4.TXT NE POSTOJI
IZLAZ:
-1
```

Primer 6

```
POKRETANJE: ./a.out
IZLAZ:
-1
```

1.1.2 Rec po rec

Zadatak 1.1.10 Napisati program koji za reč s maksimalne dužine 20 karaktera koja se zadaje sa standardnog ulaza u datoteku *rotacije.txt* upisuje sve rotacije reči s .

Jovana: p308, rec po rec, fiksno ime

Primer 1

```
INTERAKCIJA SA PROGRAMOM:
Unesite rec: abcde

ROTACIJE.TXT
abcde
bcdea
cdeab
deabc
eabcd
```

Primer 2

```
INTERAKCIJA SA PROGRAMOM:
Unesite rec: 1234

ROTACIJE.TXT
1234
2341
3412
4123
```

Primer 3

```
INTERAKCIJA SA PROGRAMOM:
Unesite rec: a=3*x+5;

ROTACIJE.TXT
a=3*x+5;
=3*x+5;a
3*x+5;a=
*x+5;a=3
x+5;a=3*
+5;a=3*x
5;a=3*x+
;a=3*x+5
```

Zadatak 1.1.11 Sa standardnog ulaza se učitava ime tekstualne datoteke i prirodan broj k . Podrazumeva se da zadata datoteka sadrži samo slova i beline i da je svaka reč iz datoteke dužine najviše 100. Program treba da učitava reči iz datoteke, da svaku reč rotira za k mesta i da tako dobijenu reč upiše u datoteku čije je ime *rotirano.txt*. Maksimalna dužina naziva datoteke je 20 karaktera.

Jovana: p3id18, rec po rec, ime jedne datoteke sa standardnog ulaza a drugo fiksno ime, nema resenje

Primer 1

```
INTERAKCIJA SA PROGRAMOM:
Unesite ime datoteke: ulaz.txt
Unesite prirodan broj: 3

ULAZ.TXT
jedan dva
tri cetiri
ROTIRANO.TXT
anjed dva tri iricet
```

Primer 2

```
INTERAKCIJA SA PROGRAMOM:
Unesite ime datoteke: in.dat
Unesite prirodan broj: 5

IN.DAT
Popodne ce biti kise
ROTIRANO.TXT
nePopod ec itib isek
```

Primer 3

```
INTERAKCIJA SA PROGRAMOM:
Unesite ime datoteke:  input.txt
Unesite prirodan broj:  0
INPUT.TXT
Popodne ce
biti kise
ROTIRANO.TXT
Popodne ce biti kise
```

Primer 4

```
INTERAKCIJA SA PROGRAMOM:
Unesite ime datoteke:  tekst.dat
Unesite prirodan broj:  7
TEKST.DAT NE POSTOJI
IZLAZ:
-1
```

Zadatak 1.1.12 Napisati program koji iz datoteke čije se ime zadaje sa standardnog ulaza prepisuje reči na standardni izlaz a one reči koje sadrže prvu reč iz datoteke i podvlaku upisuje u datoteku *rez.txt*. Maksimalna dužina naziva datoteke je 20 karaktera a reči u datoteci 50 karaktera. U slučaju greške ispisati -1.

Jovana: p3iv6, rec po rec, fiksno ime, ima resenje

Primer 1

```
INTERAKCIJA SA PROGRAMOM:
dat1.txt
DAT1.TXT
rec Opet _rec Reci rec_enica
Dva recica_
IZLAZ:
rec Opet _rec Reci rec_enica
Dva recica_
REZ.TXT:
_rec rec_enica recica_
```

Primer 2

```
INTERAKCIJA SA PROGRAMOM:
dat2.txt
DAT2.TXT
Sunce sija iznad grada
Sunce_Moje Jedan Dva Su_nce Sve Sunce123_123
suncanica.
IZLAZ:
Sunce sija iznad grada
Sunce_Moje Jedan Dva Su_nce Sve Sunce123_123
suncanica.
REZ.TXT:
Sunce_Moje Sunce123_123
```

Primer 3

```
INTERAKCIJA SA PROGRAMOM:
dat3.txt
DAT3.TXT
4 abc 1234 (5+3)*12_4 11-k
IZLAZ:
4 abc 1234 (5+3)*12_4 11-k
REZ.TXT:
(5+3)*12_4
```

Primer 4

```
INTERAKCIJA SA PROGRAMOM:
dat4.txt
DAT4.TXT NE POSTOJI
IZLAZ:
-1
```

Zadatak 1.1.13 Napisati program koji iz datoteke *razno.txt* u datoteku *palindromi.txt* prepisuje sve palindrome. Podrazumevamo da je reč palindrom ako se čita isto sa leve i desne strane bez obzira na veličinu slova. Maksimalna dužina reči je 200 karaktera a maksimalan broj reči nije poznat. U slučaju greške ispisati -1 i prekinuti izvršavanje programa.

Jovana: p3id20, rec po rec, fiksno ime, nema resenje

Primer 1

```
RAZNO.TXT
Ana i melem su primeri palindroma.
PALINDROMI.TXT:
Ana i melem
```

Primer 2

```
RAZNO.TXT
jabuka neven pomorandza kuk
Oko kapAk pero radar caj
PALINDROMI.TXT:
neven kuk
Oko kapAk radar
```

Primer 3

```
RAZNO.TXT
ovde nema palindroma
PALINDROMI.TXT:
```

Primer 4

```
RAZNO.TXT
Ana voli Milovana.
PALINDROMI.TXT:
Ana
```

Zadatak 1.1.14 U datoteci čije se ime zadaje sa standardnog ulaza nalazi se broj n ($n \leq 256$), a zatim i n reči dužine najviše 50 karaktera. Napisati program koji učitava reči iz datoteke u niz i:

(a) ispisuje ga na standardni izlaz

(b) iz niza uklanja sve duplikate i upisuje transformisani niz u datoteku *rez.txt*

U slučaju greške ispisati -1 na standardni izlaz. Maksimalna dužina naziva datoteka je 20 karaktera.

Jovana: p3iv5, rec po rec, fiksno ime, ime se unosi sa standardnog ulaza, ima resenje

Jovana: ima dve varijante resenja, koristi se u obe dvodimenzioni niz

Primer 1

```
INTERAKCIJA SA PROGRAMOM:
dat1.txt
DAT1.TXT
12 jha14 hahaha deda mraz deda
mraz deda deda jase konj konj konj
IZLAZ:
jha14 hahaha deda mraz deda mraz deda
deda jase konj konj konj
REZ.TXT:
jha14 hahaha deda mraz jase konj
```

Primer 2

```
INTERAKCIJA SA PROGRAMOM:
dat2.txt
DAT2.TXT
14
so secer supa so ljuto secer kiselo slatko
ljuto
paprika, ljuta paprika, ljuto dete
IZLAZ:
so secer supa so ljuto secer kiselo slatko
ljuto paprika, ljuta paprika, ljuto dete
REZ.TXT:
so secer supa ljuto kiselo slatko
paprika, ljuta dete
```

Primer 3

```
INTERAKCIJA SA PROGRAMOM:
  dat3.txt
DAT3.TXT
2
4 abc 1234 (5+3)*12.4 11-k
IZLAZ:
abc 1234 (5+3)*12.4 11-k
REZ.TXT:
abc 1234 (5+3)*12.4 11-k
```

Primer 4

```
INTERAKCIJA SA PROGRAMOM:
  dat4.txt
DAT4.TXT NE POSTOJI
IZLAZ:
-1
```

Zadatak 1.1.15 Napisati program koji u datoteku *izlaz.txt* prepisuje sve reči iz datoteke *ulaz.txt* čiji je zbir ASCII kodova karaktera strogo veći od 1000. Reči su odvojene prazninama i nisu duže od 200 karaktera.

Jovana: p3id19, karakter po karakter, prepisivanje, fiksno ime, nema resenje

Primer 1

```
ULAZ.TXT
Sa standardnog ulaza unosi se neoznaceni
ceo broj. Formirati novi broj koji se dobija
izbacivanjem svake druge cifre iz polaznog
broja.
IZLAZ.TXT
standardnog izbacivanjem
```

Primer 2

```
ULAZ.TXT
i sada jedan kratak primer
p1: 1234567890
p2: ABCDEFGHIJ
p3: abcdefghij
IZLAZ.TXT
abcdefghij
```

Primer 3

```
ULAZ.TXT
konstruisanje test-primera sa
i dugackim recima kao prestolonaslednik
brojevima1234567890
IZLAZ.TXT
konstruisanje test-primera
prestolonaslednik
brojevima1234567890
```

Primer 4

```
ULAZ.TXT
ima jos dugackih reci: predskazanje,
potom
nelogicnosti, zanemarivati, odugovlaciti, a ima
i i malih reci koje su kratke
predosecaj
IZLAZ.TXT
predskazanje, nelogicnosti,
zanemarivati, odugovlaciti,
predosecaj
```

1.1.3 Broj po broj

Zadatak 1.1.16 U datoteci čije se ime zadaje kao prvi argument komandne linije nalazi se prirodan broj n a zatim i n celih brojeva. Napisati program koji na standardni izlaz ispisuje koliko k -tocifrenih brojeva postoji u datoteci, pri čemu se prirodan broj k zadaje kao drugi argument komandne linije.

Jovana: v303, brojevi, argumenti

Primer 1

```
POKRETANJE: ./a.out ulaz.txt 2
ULAZ.TXT
6
15
193
-27
9790
35
1
IZLAZ:
3
```

Primer 2

```
POKRETANJE: ./a.out in.dat 5
ULAZ.TXT
4
15
193
-27
9790
IZLAZ:
0
```

Primer 3

```
POKRETANJE: ./a.out in.txt 3
ULAZ.TXT NE POSTOJI
IZLAZ:
-1
```

Primer 4

```
POKRETANJE: ./a.out in.txt
IZLAZ:
-1
```

Zadatak 1.1.17 Napisati program koji na standardni izlaz ispisuje maksimum brojeva iz datoteke *brojevi.txt*.

Jovana: p305, brojevi

Primer 1

```
BROJEVI.TXT
2.36 -16.11 5.96 8.88
-265.31 54.96 38.4
IZLAZ:
Najveci broj je: 54.96
```

Primer 2

```
BROJEVI.TXT
10.5 183.111 -90.2 3.167
IZLAZ:
Najveci broj je: 183.111
```

Primer 3

```
BROJEVI.TXT
-62.7 -190.2 -2.3 -1000
-198.25 -8
IZLAZ:
Najveci broj je: -2.3
```

Zadatak 1.1.18 Prvi red datoteke *matrice.txt* sadrži 2 cela broja manja od 50 koji predstavljaju redom broj vrsta i broj kolona realne matrice A . Svaki sledeći red sadrži po jednu vrstu matrice. Napisati program koji pronalazi sve elemente matrice A koji su jednaki zbiru svih svojih susednih elemenata i štampa ih u obliku (broj vrste, broj kolone, vrednost elementa). U slučaju greške prilikom otvaranja datoteke ispisati -1. Pretpostaviti da je sadržaj datoteke ispravan.

Jovana: p3iv4, matrice, ima resenje

Primer 1

```
MATRICE.TXT
3 4
1 2 3 4
7 2 15 -3
-1 3 1 3
IZLAZ:
(1, 0, 7)
(1, 2, 15)
```

Primer 2

```
MATRICE.TXT
2 2
1 1
-2 2
IZLAZ:
(0, 0, 1)
(0, 1, 1)
```

Primer 3

```
MATRICE.TXT
1 4
9 3 5 2
IZLAZ:
(0, 2, 5)
```

Zadatak 1.1.19 Prvi red datoteke *ulaz.txt* sadrži 2 cela broja između 2 i 50 koji predstavljaju redom broj vrsta i broj kolona realne matrice A . Svaki sledeći red sadrži po jednu vrstu matrice. Napisati program koji nalazi i štampa sve četvorke oblika $(A(i,j), A(i+1,j), A(i,j+1), A(i+1,j+1))$ u kojima su svi elementi međusobno različiti.

Jovana: p3x6, matrice, fiksno ime, nema rešenje

Jovana: preciziran tekst, uskladiti rešenje

Primer 1

```
ULAZ.TXT
3 4
1 2 3 4
7 2 15 -3
-1 3 1 3
IZLAZ:
(3, 15, 4, -3)
(7, -1, 2, 3)
(2, 3, 15, 1)
(15, 1, -3, 3)
```

Primer 2

```
MATRICE.TXT
2 2
1 1
-2 2
IZLAZ:
```

Primer 3

```
MATRICE.TXT
1 4
9 3 5 2
IZLAZ:
-1
```

1.1.4 Strukture

Jovana: Prva tri zadatka nisu rešena, trebalo bi rešiti bar prvi

Zadatak 1.1.20 U datoteci *tacke.txt* se nalazi broj tačaka, a zatim u posebnim linijama za svaku tačku njene x i y koordinate. Napisati program koji u datoteku *rastojanja.txt* upisuje rastojanje svake od učitanih tačaka od koordinatnog početka, a na standardni izlaz koordinate tačke koja je od njega najudaljenija. Koristiti strukturu *Tacka* sa poljima x i y , kao i funkciju kojom se računa rastojanje. Pretpostaviti da broj tačaka u datoteci neće biti veći od 50.

Jovana: p307, strukture, fiksno ime

Primer 1

```
TACKE.TXT
4
11 -2
3 5
8 -8
0 4

RASTOJANJA.TXT
11.18
5.29
11.31
4.00

IZLAZ:
Najudaljenija je tačka: 8 -8
```

Primer 2

```
TACKE.TXT
-2
0 0
9 -8

IZLAZ:
Greska: Nedozvoljen broj tacaka!
```

Zadatak 1.1.21 Data je struktura kojom se opisuje trodimenzioni vektor:

```
typedef struct{
    int x;
    int y;
    int z;
} vektor;
```

U datoteci *vektori.txt* nalazi se nepoznati broj vektora (najviše 200). Napisati program koji učitava vektore iz ove datoteke u niz i ispisuje na standardni izlaz koordinate vektora sa najvećom dužinom. Dužina vektora se izračunava po formuli:

$$|v| = \sqrt{x^2 + y^2 + z^2}$$

U slučaju greške ispisati -1.

Jovana: p3x5, struktura, fiksno ime, nema resenje

Primer 1

```
VEKTORI.TXT
2
4 -1 7
3 1 2
IZLAZ:
4 -1 7
```

Primer 2

```
VEKTORI.TXT
67
IZLAZ:
-1
```

Primer 3

```
VEKTORI.TXT
3
0 0 0
0 1 0
1 0 0
IZLAZ:
0 1 0
```

Primer 4

```
|| VEKTORI.TXT  
|| 4  
|| 3 0 1  
|| 4 5 2  
|| 1 0 0  
|| 2 -1 2  
|| IZLAZ:  
|| 4 5 2
```

Primer 5

```
|| VEKTORI.TXT  
|| 4  
|| 3 0 1  
|| 4 5 2  
|| 1  
|| 2 -1 2  
|| IZLAZ:  
|| -1
```

Primer 6

```
|| VEKTORI.TXT  
|| IZLAZ:  
|| -1
```

Zadatak 1.1.22 Data je struktura koja opisuje pravougaonik dužinama svojih stranica i imenom:

```
typedef struct{  
    unsigned int a, b;  
    char ime[5];  
}_pravougaonik;
```

Napisati program koji iz datoteke čije ime se zadaje kao argument komandne linije učitava podatke o pravougaonicima (nepoznato koliko), a zatim ispisuje imena onih pravougaonika koji su kvadrati i vrednost najveće površine među pravougaonicima koji nisu kvadrati. U slučaju unosa nekorektnih dužina stranica pravougaonika ili nekorektne vrednosti broja n , ispisati -1. Maksimalan broj pravougaonika je 200.

Jovana: p3x1, strukture, argumenti, nema resenja

Primer 1

```
|| POKRETANJE: ./a.out  
|| pravougaonici.dat  
|| PRAVOUGAONICI.DAT  
|| 2 4 p1  
|| 3 3 p2  
|| 1 6 p3  
|| IZLAZ:  
|| p2 8
```

Primer 2

```
|| POKRETANJE: ./a.out dva.dat  
|| DVA.DAT  
|| 5 2 pm  
|| 4 7 pv  
|| IZLAZ:  
|| 28
```

Primer 3

```
|| POKRETANJE: ./a.out tri.dat  
|| TRI.DAT  
|| 5 5 m  
|| 3 3 s  
|| 8 8 xl  
|| IZLAZ:  
|| m s xl
```

Primer 4

```
|| POKRETANJE: ./a.out  
|| primerx.dat  
|| PRIMERX.DAT  
|| 9 7 p  
|| IZLAZ:  
|| 63
```

Primer 5

```
|| POKRETANJE: ./a.out empty.dat  
|| EMPTY.DAT  
|| IZLAZ:
```

Primer 6

```
|| POKRETANJE: ./a.out  
|| rectangles.txt  
|| RECTANGLES.TXT NE POSTOJI  
|| IZLAZ:  
|| -1
```

Zadatak 1.1.23 U prvom redu datoteke *studenti.txt* se nalazi broj studenata, a zatim u posebnim linijama za svakog studenta korisničko ime na Alasu i poslednjih pet ocena koje je dobio. Napisati program koji pronalazi studenta koji je ostvario najbolji uspeh i ispisuje njegove podatke. Ukoliko više studenata ima maksimalni prosek, ispisati prvog. Pretpostaviti da broj studenata neće biti veći od 100.

Jovana: p306, strukture, fiksno ime

Primer 1

```
STUDENTI.TXT
5
mr15239 10 9 9 8 10
mi14005 8 8 9 8 10
ml15112 9 8 8 7 10
mr15007 10 10 10 10 10
mn13208 7 7 9 6 10

IZLAZ:
korisnicko ime: mr15007, prosek ocena: 10.00
```

Primer 2

```
STUDENTI.TXT
3
mr16156 10 9 9 10 10
mi17234 9 9 10 10 10
ml17084 9 8 8 8 8

IZLAZ:
korisnicko ime: mr16156, prosek ocena: 9.6
```

Zadatak 1.1.24 Kreirati strukturu *Student* koja sadrži:

- *ime_i_prezime* (u polju se čuva ime i prezime studenta, napr. "Marko Markovic", maksimalna dužina polja je 100 karaktera),
- *oc* (sadrži najviše 10 ocena studenta)
- *br_ocena* (ukupan broj ocena za studenata)
- *pr_oc* (prosečna ocena)

U datoteci čije se ime zadaje kao argument komandne linije se nalaze podaci o studentima. Za svakog studenta dato je ime, prezime i niz ocena koji se završava nulom. Svi podaci su razdvojeni razmacima. Napisati program koji učitava podatke o studentima i na standardni izlaz ispisuje podatke za studenta sa najvećim prosekom (prosek ispisati na 2 decimale). Maksimalan broj studenta je 100. UPUTSTVO: Ime i prezime studenta se mogu pročitati pomoću specifikatora %s a potom se za kreiranje niske ime_i_prezime u traženom formatu može iskoristiti funkcija strcat.

Jovana: p3x3, struktura, argumenti, nema resenja

Primer 1

```
|| POKRETANJE: ./a.out
|| studenti.txt
|| STUDENTI.TXT
|| Marko Markovic 5 6 7 8 9 0
|| Jelena Jankovic 10 10 10 0
|| Filip Viskovic 10 9 8 7 6 0
|| Jana Peric 10 10 9 9 8 8 7 0
|| IZLAZ:
|| Jelena Jankovic 10 10 10 0
|| 10.00
```

Primer 2

```
|| POKRETANJE: ./a.out
|| IZLAZ:
|| -1
```

Primer 3

```
|| POKRETANJE: ./a.out
|| IZLAZ:
|| -1
```

Zadatak 1.1.25 Imena ulazne i izlazne datoteke se redom navode kao argumenti komandne linije. U ulaznoj datoteci se nalaze podaci o razlomcima: u prvom redu se nalazi broj razlomaka, a u svakom sledećem redu brojilac i imenilac jednog razlomka. Potrebno je kreirati strukturu koja opisuje razlomak i učitati niz razlomaka iz datoteke, a potom:

- (a) ukoliko je navedena opcija x , upisati u izlaznu datoteku recipročni razlomak za svaki razlomak iz niza (npr. za $2/3$ treba upisati $3/2$)
- (b) ukoliko je navedena opcija y , upisati u izlaznu datoteku realnu vrednost recipročnog razlomka svakog razlomka iz niza (npr. za $2/3$ treba upisati 1.5)

Pretpostaviti da se u ulaznoj datoteci nalazi najviše 100 razlomaka. U slučaju greške, na standardni izlaz upisati -1.

Jovana: v305, strukture, argumenti, opcije

Primer 1

```
|| POKRETANJE: ./a.out ulaz.txt izlaz.txt -x
|| ULAZ.TXT
|| 4
|| 1 5
|| 19 3
|| -2 7
|| 97 90
|| IZLAZ.TXT
|| 5/1
|| 3/19
|| -7/2
|| 90/97
```

Primer 2

```
|| POKRETANJE: ./a.out ulaz.txt izlaz.txt -y
|| ULAZ.TXT
|| 4
|| 1 5
|| 19 3
|| -2 7
|| 97 90
|| IZLAZ.TXT
|| 5.000000
|| 0.157894
|| -3.500000
|| 0.927835
```

Primer 3

```
|| POKRETANJE: ./a.out ulaz.txt izlaz.txt -y
|| ULAZ.TXT NE POSTOJI
|| IZLAZ:
|| -1
```

Primer 4

```
|| POKRETANJE: ./a.out ulaz.txt izlaz.txt
|| IZLAZ:
|| -1
```

Zadatak 1.1.26 Za svaki automobil poznati su marka, model i cena. Iz datoteke čije se ime zadaje sa standardnog ulaza učitava se broj automobila a potom i podaci za svaki automobil. Program treba da:

- (a) ispiše prosečnu cenu po marki kola
- (b) za maksimalnu cenu koju je kupac spreman da plati, a koja se zadaje kao argument komandne linije, da ispiše automobile u tom cenovnom rangi zajedno sa prosečnom cenom odgovarajuće marke

Pretpostaviti da se model i marka sastoje od jedne reči i da svaka od njih sadrži najviše 30 karaktera kao i da se u datoteci nalaze podaci za najviše 100 automobila.

Jovana: v306, strukture, argumenti

<i>Primer 1</i>	<i>Primer 2</i>	<i>Primer 3</i>
<pre> POKRETANJE: ./a.out 4000 INTERAKCIJA SA PROGRAMOM: dat1.txt DAT1.TXT 7 renault twingo 2900 renault megan 6250 renault clio 3650 dacia logan 5400 dacia sandero 7800 fiat bravo 4900 fiat linea 4290 IZLAZ: Informacije o prosečnoj ceni po markama: renault 4266.67 3 dacia 6600.00 2 fiat 4595.00 2 Kola u vasem cenovnom rangi: renault twingo 4266.67 renault clio 4266.67 </pre>	<pre> POKRETANJE: ./a.out 5000 INTERAKCIJA SA PROGRAMOM: dat1.txt DAT1.TXT NE POSTOJI IZLAZ: -1 </pre>	<pre> POKRETANJE: ./a.out IZLAZ: -1 </pre>

1.1.5 Linija po linija

Zadatak 1.1.27 Napisati program koji u datoteci čije se ime navodi kao argument komandne linije određuje liniju maksimalne dužine i ispisuje je na standardni izlaz. Ukoliko ima više takvih linija, ispisati onu koja je leksikografski prva. Pretpostaviti da su linije maksimalne dužine 80 karaktera.

Jovana: v302, linije, dužina, argumenti

Primer 1

```
POKRETANJE: ./a.out test.txt
TEST.TXT
Danas je veoma hladno decembarsko
popodne. Ne pada sneg, kazu mozda
ce sutra.

IZLAZ:
Danas je veoma hladno decembarsko
```

Primer 2

```
POKRETANJE: ./a.out in.txt
IN.TXT NE POSTOJI

IZLAZ:
-1
```

Primer 3

```
POKRETANJE: ./a.out
IZLAZ:
-1
```

Zadatak 1.1.28 Kao argumenti komandne linije se zadaju ime datoteke i ceo broj k . Napisati program koji na standardni izlaz ispisuje sve linije zadate datoteke čija je dužina veća od k . Možemo pretpostaviti da dužina linije neće biti veća od 80 karaktera.

Jovana: p303, dužina, argumenti, linije

Primer 1

```
POKRETANJE: ./a.out test.txt 7
TEST.TXT
Teme koje su obradjivane:
Petlje
Funkcije
Nizovi
Strukture

IZLAZ:
Teme koje su obradjivane:
Funkcije
Strukture
```

Primer 2

```
POKRETANJE: ./a.out test.txt
IZLAZ:
Greska: Pogresan broj argumenata!
```

Zadatak 1.1.29 U datoteci čije se ime navodi kao prvi argument komandne linije navedena je reč r i niz linija. Napisati program koji u datoteku čije se ime navodi kao drugi argument komandne linije upisuje sve linije u kojima se reč r pojavljuje bar n puta gde je n prirodan broj koji se unosi sa standardnog ulaza. Računaju se i samostalna pojavljivanja reči r i pojavljivanja u okviru neke druge reči. Ispis treba da bude u formatu broj_pojavljivanja:linija.

Jovana: v304, linije, podniska, argumenti

Primer 1

```
POKRETANJE: ./a.out input.txt output.txt
INTERAKCIJA SA PROGRAMOM:
  Unesite prirodan broj: 2
INPUT.TXT
  sto
  stolica lampa
  postotak Stopiranje stopa
  presto Ostoja stotina prostorija
OUTPUT.TXT
  2: postotak Stopiranje stopa
  4: presto Ostoja stotina prostorija
```

Primer 2

```
POKRETANJE: ./a.out input.txt output.txt
INTERAKCIJA SA PROGRAMOM:
  Unesite prirodan broj: 3
INPUT.TXT
  red
  redar za ovu nedelju
  redosled ured
  odrediti raspored
OUTPUT.TXT
```

Primer 3

```
POKRETANJE: ./a.out in.txt out.txt
IN.TXT NE POSTOJI
INTERAKCIJA SA PROGRAMOM:
  -1
```

Primer 4

```
POKRETANJE: ./a.out in.txt
INTERAKCIJA SA PROGRAMOM:
  -1
```

Zadatak 1.1.30 Napisati program koji prebrojava koliko se linija datoteke *ulaz.txt* završava niskom *s* koja se učitava sa standardnog ulaza. Može se postaviti da dužina linije neće biti veća od 80 karaktera, kao i da dužina niske *s* neće biti veća od 20 karaktera.

Jovana: p304, linije, podniska

Primer 1

```
ULAZ.TXT
  abcde abcde
  abcde aab
  abcde abcde abcde
  abcde abcde Aab
  abcde abcde ab
  abcde abcde abcde abcde
INTERAKCIJA SA PROGRAMOM:
  Unesite nisku s: ab
  Broj linija: 3
```

Primer 2

```
ULAZ.TXT
  abcde abcde
  abcde
  abcde abcde AB
INTERAKCIJA SA PROGRAMOM:
  Unesite nisku s: ab
  Broj linija: 0
```

Zadatak 1.1.31 Napisati program koji linije koje se učitavaju sa standardnog ulaza sve do kraja ulaza prepisuje u datoteku *izlaz.txt* i to, ako je prilikom pokretanja zadata opcija *-v* ili *-V* samo one linije koje počinju velikim slovom, ako je zadata opcija *-m* ili *-M* samo one linije koje počinju malim slovom, a ako je opcija izostavljena sve linije. Pretpostaviti da linije neće biti duže od 80 karaktera.

Jovana: p309, linije, prepisivanje, opcije, fiksno ime

Primer 1

```
POKRETANJE: ./a.out -m
INTERAKCIJA SA PROGRAMOM:
  Unesite recenice:
    programiranje u C-u je
      zanimljivo
    Volim programiranje!
    Kada porastem bicu programer!
    u slobodno vreme programiram

IZLAZ.TXT
  programiranje u C-u je
    zanimljivo
  u slobodno vreme programiram
```

Primer 2

```
POKRETANJE: ./a.out -V
INTERAKCIJA SA PROGRAMOM:
  Unesite recenice:
    programiranje u C-u je
      zanimljivo
    Volim programiranje!
    Kada porastem bicu programer!
    u slobodno vreme programiram

IZLAZ.TXT
  Volim programiranje!
  Kada porastem bicu programer!
```

Primer 3

```
POKRETANJE: ./a.out -k
INTERAKCIJA SA PROGRAMOM:
  Greska: Pogresno pokretanje
    programa!
```

Zadatak 1.1.32 Napisati program koji poredi dve datoteke i ispisuje redni broj linija u kojima se datoteke razlikuju. Imena datoteka se zadaju kao argumenti komandne linije. U slučaju neuspešnog otvaranja datoteka ispisati poruku o grešci. Pretpostaviti da je maksimalna dužina reda u datoteci 200 karaktera. Ukoliko nisu zadati potrebni argumenti komandne linije ispisati poruku o grešci. Linije brojati počevši od 1.

Jovana: iv10, linije, poredjenje, argumenti, ima resenje

Primer 1

```
POKRETANJE: ./a.out ulaz.txt izlaz.txt
ULAZ.TXT
  danas vezbamo
  programiranje
  ovo je primer kad su
  datoteke iste
IZLAZ.TXT:
  danas vezbamo
  programiranje
  ovo je primer kad su
  datoteke iste
IZLAZ:
```

Primer 2

```
POKRETANJE: ./a.out primer1.dat primer2.dat
PRIMER1.DAT
  danas vezbamo
  analizu
  ovo je primer kad
  su datoteke razlicite
PRIMER2.DAT
  danas vezbamo
  programiranje
  ovo je primer kad su
  datoteke razlicite
IZLAZ:
  2 3 4
```

Primer 3

```
POKRETANJE: ./a.out prva.dat druga.dat
PRVA.DAT
ovo je primer
kada su
datoteke
razlicite duzine
DRUGA.DAT
kada su
programiranje
datoteke
razlicite
duzine
i kada treba ispisati broj
tih redova
IZLAZ:
1 4 5 6 7
```

Primer 4

```
POKRETANJE: ./a.out prva.dat
IZLAZ:
greska
```

1.2 Rešenja