

Programiranje 2

Razni zadaci - datoteke, strukture

1 ZADACI

Zadatak 1 Napisati program koji prepisuje datoteku `ulaz.txt` u datoteku `izlaz.txt` i to:

- a) karakter po karakter
- b) liniju po liniju

U slučaju greške ispisati `-1`. Greškom se smatra neuspešno otvaranje jedne ili druge datoteke.

Primer 1: 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: ulaz.txt: Cena soka je 30 Cena vina je 150 Cena limunade je 200 Cena sendvica je 120 izlaz.txt: Cena soka je 30 Cena vina je 150 Cena limunade je 200 Cena sendvica je 120	Primer 3: ulaz.txt: izlaz.txt: Cena soka je 30 Cena vina je 150 Cena limunade je 200 Cena sendvica je 120
---	---	--

Zadatak 2 Napisati program koji na standardni izlaz ispisuje broj linija u tekstualnom fajlu sa imenom `knjiga.txt`. U slučaju greške u otvaranju i zatvaranju datoteka, prijaviti odgovarajući komentar na `stdout` ili `stderr`.

Primer 1: ulaz.txt: danas je lep dan i ja zelim da postanem programer 3	Primer 2: ulaz.txt: Cena soka je 30 Cena vina je 150 Cena limunade je 200 Cena sendvica je 120 4	Primer 3: ulaz.txt: 0
---	---	--

Zadatak 3 Sastaviti program koji sa standardnog ulaza učitava imena dve datoteke (ulazna i izlazna datoteka) i iz ulazne datoteke kopira u izlaznu svaki drugi karakter polazeći od prvog pročitanoog karaktera. U slučaju greške u otvaranju i zatvaranju datoteka, prijaviti odgovarajući komentar na `stdout` ili `stderr`.

Primer 1: ulaz.txt izlaz.txt ulaz.txt: danas je lep dan i ja zelim da postanem programer	Primer 2: prva.dat druga.dat prva.dat: Cena soka je 30 Cena vina je 150 Cena limunade je 200 Cena sendvica je 120	Primer 3: primer.c prazna.txt primer.c: #include <stdio.h> int main() { } }
--	--	---

izlaz.txt: aa elpdnij ei apsae rgae	druga.dat: eask e3 eavn e10Cn iuaej 0 easnvc e10	prazna.txt: icue<ti.>itmi(
---	---	-------------------------------

Zadatak 4 Sa standardnog ulaza učitavaju se imena dve tekstualne datoteke i jedan karakter. Napisati program koji prepisuje datoteku čije se ime navodi kao prvo u datoteku čije ime se navodi kao drugo. Ukoliko je učitani karakter u program prilikom prepisivanja treba da zamenjuje sva mala slova velikim, a ukoliko je učitani karakter 1 sva velika slova se zamenjuju malim. U slučaju greske ispisati -1. Greška može biti neuspešno otvaranje datoteke ili pogrešno zadat karakter

Primer 1: 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: prva.dat druga.dat 1 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: primer.c prazna.txt V primer.c: #include <stdio.h> int main() { } -1
---	---	--

Zadatak 5 Sastaviti program koji sa standardnog ulaza prima ime datoteke koju treba otvoriti. Ispisati (na standardnom izlazu) koja cifra (među svim ciframa koje se pojavljuju u datoteci) ima najveći broj pojavljivanja. U slučaju greske pri otvaranju datoteke ispisati -1. Ukoliko nema cifara u datoteci ispisati -1.

Primer 1: ulaz.txt ulaz.txt: danas je lep dan i Ja zelim da postanem programer -1	Primer 2: prva.dat prva.dat: Cena soka je 30 Cena vina je 150 Cena limunade je 200 Cena sendvica je 120 0	Primer 3: primer.c primer.c: #include <stdio.h> int main() { } -1
--	---	---

Zadatak 6 Sa standardnog ulaza se učitava prirodan broj k i ime datoteke u kojoj se prvo nalazi prirodan broj n a zatim i n celih brojeva. Napisati program koji prebrojava koliko k -tocifrenih brojeva postoji u datoteci. U slučaju greske prilikom otvaranja datoteke ispisati -1. Pretpostaviti da je sadržaj datoteke ispravan.

Primer 1: 3 ulaz.txt ulaz.txt: 10 1 9 20 400 708 -2 -520 1000 403 20000 4	Primer 2: 1 prva.dat prva.dat: 4 1 20 9 -8 3	Primer 3: 5 primer.c primer.c: 3 4 5 50000 1
---	---	--

Zadatak 7 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.

```
Primer1:
3 4
1 2 3 4
7 2 15 -3
-1 3 1 3

(1, 0, 7)
(1, 2, 15)
```

Zadatak 8 Napisati program koji za dve datoteke čija su imena data kao prvi i drugo na standardnom ulazu, radi sledeće: za cifru u prvoj datoteci, u drugu datoteku se upisuje 0, za slovo se upisuje 1, a za sve ostale karaktere se upisuje 2.

```
Primer1:
prva.dat druga.dat
prva.dat:
Cena soka je 30
Cena vina je 150
Cena limunade je 200
Cena sendvica je 120

druga.dat:
1111211112112002111121111211200021111211111111112112000211112111111112112000
```

Zadatak 9 Ako je data tekstualna datoteka plain.txt napraviti tekstualnu datoteku sifra.txt tako što se svako slovo zamenjuje svojim prethodnikom (ciklično) suprotne velicine b sa A, B sa a, a sa Z, A sa z, itd. Podrazumevati da se na sistemu koristi tabela karaktera ASCII.

Zadatak 10 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.

Zadatak 11 Napisati program koji u datoteku izlaz.txt prepisuje sve reči iz datoteke ulaz.txt čiji je zbir ascii kodova slova strogo veći od 1000. Reči su odvojene prazninama i nisu duže od 200 karaktera.

```
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:
konstruisanje test-primera sa
i dugackim recima kao prestolonaslednik
brojevima1234567890

izlaz.txt:
konstruisanje test-primera
prestolonaslednik
brojevima1234567890
```

```
Primer 3:
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
```

```
Primer 4:
ulaz.txt:
i sada jedan kratak primer
p1: 1234567890
p2: ABCDEFGHIJ
p3: abcdefghij
```

```
izlaz.txt:
abcdefghij
```

Zadatak 12 U datoteci `razno.txt` nalazi se tekst. U datoteku `palindromi.txt` prepisati sve reči iz datoteke `razno.txt` koje su palindromi. Reč je palindrom ako se čita isto sa leve i desne strane. Za reč smatramo niz karaktera koji se nalazi između belina i koji nije duži od 200 karaktera. Dozvoljeno je korišćenje specifikatora za čitanje reči. Maksimalan broj reči nije poznat. U slučaju greške ispisati -1 i prekinuti izvršavanje programa.

Primer 1: <code>razno.txt:</code> Ana i melem su primeri palindroma. <code>palindromi.txt:</code> Ana i melem	Primer 2: <code>razno.txt:</code> jabuka neven pomorandza kuk <code>palindromi.txt:</code> neven kuk
Primer 3: <code>razno.txt:</code> Kajak voda teret PoTop <code>palindromi.txt:</code> Kajak teret PoTop	Primer 4: <code>razno.txt:</code> Oko kapAk pero radar caj <code>palindromi.txt:</code> Oko kapAk radar

Zadatak 13 U datoteci čije se ime navodi na standardnom ulazu programa nalazi se broj `n`, a zatim `i` `n` reči (dužine najviše 50 karaktera). Napisati program koji učitava ovaj niz i

1. ispisuje ga [3],

2. iz njega uklanja sve duplikate i u datoteku `rez.txt` ispisuje transformisani niz [4]

U slučaju greške ispisati -1.

Primer 1: <code>dat1.txt</code> <code>dat1.txt: 12 jha14 hahaha deda mraz deda</code> mraz deda deda jase konj konj konj <code>jha14 hahaha deda mraz deda mraz deda</code> <code>deda jase konj konj konj</code> <code>rez.txt: jha14 hahaha deda mraz jase konj</code>	Primer 2: <code>dat2.txt</code> <code>dat2.txt: 14</code> so secer supa so ljuto secer kiselo slatko ljuto paprika, ljuta paprika, ljuto dete so secer supa so ljuto secer kiselo slatko ljuto paprika, ljuta paprika, ljuto dete <code>rez.txt: so secer supa ljuto kiselo slatko</code> paprika, ljuta dete
Primer 3: <code>dat3.txt</code> <code>dat.txt: 17 Buducnost televizije su ultra HD, odnosno 4K</code> uredaji koji imaju ogromnu dijagonalu ekrana i znacajno vise piksela Buducnost televizije su ultra HD, odnosno 4K uredaji koji imaju ogromnu dijagonalu ekrana i znacajno vise piksela <code>rez.txt: Buducnost televizije su ultra HD, odnosno 4K</code> uredaji koji imaju ogromnu dijagonalu ekrana i znacajno vise piksela	

Zadatak 14 U datoteci čije se ime navodi na standardnom ulazu programa nalazi se broj `n`, a zatim `i` `n` reči (dužine najviše 50 karaktera). Napisati program koji učitava ovaj niz i

1. ispisuje ga, [3]

2. u datoteku `rez.txt` upisuje sve reči koje sadrže prvu reč i podvlaku. [4]

U slučaju greške ispisati -1.

Primer 1: dat1.txt dat1.txt: 7 rec Opet _rec Reci rec_enica DVa recica_ rec Opet _rec Reci rec_enica DVa recica_ rez.txt: _rec rec_enica recica_		Primer 2: dat2.txt dat2.txt: 11 Sunce sija iznad grada Sunce_Moje Jedan Dva Su_nce Sve Sunce123_123 suncanica. Sunce sija iznad grada Sunce_Moje Jedan Dva Su_nce Sve Sunce123_123 suncanica. rez.txt: Sunce_Moje Sunce123_123

Primer 3: dat3.txt dat.txt: 18 Na danasnji dan roden je poznati engleski pisac Carls Dikens, a umro reformator srpskog jezika Vuk Stefanovic Karadzic. Na danasnji dan roden je poznati engleski pisac Carls Dikens, a umro reformator srpskog jezika Vuk Stefanovic Karadzic. rez.txt:		

Zadatak 15 Imena dve datoteke se zadaje na standardnom ulazu. U prvoj datoteci navedena je rec **r** i niz linija. Napisati program koji u drugu datoteku upisuje sve linije u kojima se rec **r** pojavljuje bar **n** puta, gde je **n** prirodan broj koji se unosi sa standardnog ulaza. Ispis treba da bude u formatu broj-pojavljivanja: linija. Linije brojati počevši od 1.

Zadatak 16 Program se pokreće tako sto se navedu nazivi dve datoteke (ulazna i izlazna) i opcije. U datoteci čije se ime navodi kao prvi argument komandne linije nalaze se 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 drugu datoteku recipročni razlomak za svaki razlomak iz niza (npr. za $2/3$ treba upisati $3/2$)
- b) ukoliko je navedena opcija **y**, upisati u datoteku realnu vrednost (ispisati samo ne-nula decimale) recipročnog razlomka svakog razlomka iz niza (npr. za $2/3$ treba upisati 1.5)

Možemo pretpostaviti da se u datoteci sa podacima o razlomcima nalazi najviše 100 razlomaka.

Prilikom pokretanja programa se, pored naziva ulazne i izlazne datoteke, navode i opcije **-x** i **-y**. Moguće je navesti jednu ili obe opcije, sto znači da je minimalni broj argumenata 3.

Mogući načini pokretanja:

```
./a.out ulaz.txt izlaz.txt -x
./a.out ulaz.txt izlaz.txt -y
./a.out ulaz.txt izlaz.txt -yx
./a.out ulaz.txt izlaz.txt -xy
```

Zadatak 17 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) izračuna 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 rangu zajednu sa prosečnom cenom odgovarajuće marke.

Možemo 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.