

Programiranje 2, i smer, Kolokvijum 2018., I grupa

Na Desktop-u napraviti direktorijum čije je ime u formatu InicijaliAsistenta_ImePrezime_BrojIndeksa_1. Sve zadatke sačuvati u ovom direktorijumu. Zadatke imenovati sa 1.c, 2.c i 3.c.

Na primer, student Petar Marković, čiji je broj indeksa 45/2017 i kome je asistent Milica Selaković treba da napravi folder MS_PetarMarkovic_mi17045_1.

1. Kao argument komandne linije se zadaje ime datoteke i niska. Na standardni izlaz ispisati redne brojeve linija u kojima se data niska pojavljuje i indeks prvog pojavljivanja niske u toj liniji ili 0 ukoliko nema pojavljivanja niske u datoteci. Brojanje linija početi od 1. Svaka linija sadrži maksimalno 100 karaktera.

U slučaju greške na standardni izlaz za greške ispisati -1 .

Primer 1

```
POZIV: /a.out ulaz.txt kis
ULAZ.TXT
Danas je suncano i ne pada kisa
medjutim sutra ce mozda padati
ako bude padala
bice nam potreban kisobran
kisa nije padala prethodnih dana
STANDARDNI IZLAZ:
1 27
4 18
5 0
```

Primer 2

```
POZIV: /a.out ulaz.txt programiranje
ULAZ.TXT
Danas je suncano i ne pada kisa
medjutim sutra ce mozda padati
ako bude padala
bice nam potreban kisobran
STANDARDNI IZLAZ:
0
```

Primer 3

```
POZIV: ./a.out
STANDARDNI IZLAZ ZA GREŠKE:
-1
```

Primer 4

```
POZIV: ./a.out ulaz.txt programiranje
Datoteka ulaz.txt ne postoji
STANDARDNI IZLAZ ZA GREŠKE:
-1
```

2. Napisati rekurzivnu funkciju `int podeliCifreSa3(int n)` kojom se za dati prirodan broj n određuje broj dobijen deljenjem sa 3 svake cifre broja n koja je deljiva sa 3.

Napisati program koji testira ovu funkciju, tako što sa standardnog ulaza učitava ceo broj n , i na standardni izlaz ispisuje vrednost funkcije `podeliCifreSa3(n)`. U slučaju greške ispisati -1 na standardni izlaz za greške.

Napomena: Zadatak mora biti uradjen rekurzivno. Nije dozvoljeno korišćenje statičkih i globalnih promenljivih, menjanje zaglavlja funkcije i pisanje pomoćnih funkcija.

Primer 1

```
INTERAKCIJA SA PROGRAMOM:
STANDARDNI ULAZ:
123456789
STANDARDNI IZLAZ:
121452783
```

Primer 2

```
INTERAKCIJA SA PROGRAMOM:
STANDARDNI ULAZ:
3003
STANDARDNI IZLAZ:
1001
```

Primer 3

```
INTERAKCIJA SA PROGRAMOM:
STANDARDNI ULAZ:
-123
STANDARDNI IZLAZ ZA GREŠKE:
-1
```

Primer 4

```
INTERAKCIJA SA PROGRAMOM:
STANDARDNI ULAZ:
0
STANDARDNI IZLAZ ZA GREŠKE:
-1
```

3. U fajlu `arhiva.txt` se nalaze podaci o prihodima neke kompanije. Svaka linija datoteke sadrži sledeće informacije: godina prihod rashod. Godina je ceo pozitivan broj. Prihod i rashod su realni nenegativni brojevi. Nije poznato koliko linija ima u datoteci. Datoteka je sortirana rastuće po godini. Sa standardnog ulaza učitava se godina. Na standardni izlaz ispisati zaradu (ili gubitak) koji je kompanija ostvarila te godine. Zarada/gubitak se računa kao razlika prihoda i rashoda. Na slučaj kada je razlika 0 gledati kao da je u pitanju zarada. Ukoliko ne postoje informacije za zadatak godinu ispisati poruku `Ne postoje informacije`. U slučaju greške na standardni izlaz za greške ispisati -1 .

Napomena: Zadatak mora biti rešen korišćenjem binarne pretrage i dinamičke alokacije memorije.

Primer 1

```
ARHIVA.TXT
1950 230000 130000
1951 0 0
1952 0 0
1953 90000 30000
1954 45000 90000
STANDARDNI ULAZ:
1953
STANDARDNI IZLAZ:
Zarada: 60000
```

Primer 2

```
ARHIVA.TXT
1950 230000 130000
1951 0 0
1952 0 0
1953 90000 30000
1954 45000 90000
STANDARDNI ULAZ:
1954
STANDARDNI IZLAZ:
Gubitak: 45000
```

Primer 3

```
ARHIVA.TXT
2016 -2000 5000
-1000 1000 1000
STANDARDNI IZLAZ ZA GREŠKE:
-1
```

Primer 4

```
ARHIVA.TXT
1950 230000 130000
1951 0 0
1952 0 0
1953 90000 30000
1954 45000 90000
STANDARDNI ULAZ:
2004
STANDARDNI IZLAZ:
Ne postoje informacije
```

Programiranje 2, i smer, Kolokvijum 2018., II grupa

Na Desktop-u napraviti direktorijum čije je ime u formatu InicijaliAsistenta_ImePrezime_BrojIndeksa_2. Sve zadatke sačuvati u ovom direktorijumu. Zadatke imenovati sa 1.c, 2.c i 3.c.

Na primer, student Petar Marković, čiji je broj indeksa 45/2017 i kome je asistent Milica Selaković treba da napravi folder MS_PetarMarkovic_mi17045_2.

1. Kao prvi argument komandne linije se zadaje opcija, a kao drugi broj elemenata niza n . Sa standardnog ulaza se učitava niz celih brojeva dimenzije n . Ukoliko je zadata opcija `-p` u datoteku `izlaz.txt` upisati proizvod svih brojeva, a ukoliko je zadata opcija `-s` u datoteku `izlaz.txt` upisati njihov zbir.
U slučaju greške na standardni izlaz za greške ispisati `-1`.

Primer 1

```
POZIV: /a.out -p 5
STANDARDNI ULAZ:
1 2 3 4 5
IZLAZ.TXT
120
```

Primer 2

```
POZIV: /a.out -s 10
STANDARDNI ULAZ:
1 2 3 4 5 6 7 8 9 10
IZLAZ.TXT
55
```

Primer 3

```
POZIV: ./a.out -s -1
STANDARDNI IZLAZ ZA GREŠKE:
-1
```

Primer 4

```
POZIV: ./a.out -q 10
STANDARDNI IZLAZ ZA GREŠKE:
-1
```

2. Napisati rekurzivnu funkciju `int ukloniParneCifreNeparneUmanji(int n)` kojom se za dati prirodan broj n određuje broj dobijen tako što se svaka parna cifra broja n ukloni iz broja, a svaka neparna cifra umanjuje za 1. Napisati program koji testira ovu funkciju, tako što sa standardnog ulaza učitava ceo broj n , i na standardni izlaz ispisuje vrednost funkcije `ukloniParneCifreNeparneUmanji(n)`.
U slučaju greške ispisati `-1` na standardni izlaz za greške.

Napomena: Zadatak mora biti urađen rekurzivno. Nije dozvoljeno korišćenje statičkih i globalnih promenljivih, menjanje zaglavlja funkcije i pisanje pomoćnih funkcija.

Primer 1

```
INTERAKCIJA SA PROGRAMOM:
STANDARDNI ULAZ:
123456789
STANDARDNI IZLAZ:
2468
```

Primer 2

```
INTERAKCIJA SA PROGRAMOM:
STANDARDNI ULAZ:
2222
STANDARDNI IZLAZ:
0
```

Primer 3

```
INTERAKCIJA SA PROGRAMOM:
STANDARDNI ULAZ:
1000
STANDARDNI IZLAZ:
0
```

Primer 4

```
INTERAKCIJA SA PROGRAMOM:
STANDARDNI ULAZ:
-123
STANDARDNI IZLAZ ZA GREŠKE:
-1
```

Primer 5

```
INTERAKCIJA SA PROGRAMOM:
STANDARDNI ULAZ:
0
STANDARDNI IZLAZ ZA GREŠKE:
-1
```

Primer 6

```
INTERAKCIJA SA PROGRAMOM:
STANDARDNI ULAZ:
5000
STANDARDNI IZLAZ:
4
```

3. U datoteci `rezultati.txt` se nalaze rezultati nekog ispita. Svaka linija datoteke sadrži sledeće podatke: ime prezime broj_poena. Ime i prezime su niske maksimalne duzine 50 karaktera. Broj poena je nenegativan realan broj. Datoteka je sortirana po broju poena rastuće. Nije poznato koliko studenata je izašlo na ispit. Sa standardnog ulaza se unosi jedan realan broj x . Na standardni izlaz ispisati podatke o studentu koji je na ispitu osvojio x poena. Ukoliko ima više takvih studenata, ispisati podatke o bilo kom. Ukoliko ne postoji student koji je osvojio x poena ispisati poruku `Niko nije ostvario zadati broj poena`.
U slučaju greške na standardni izlaz za greške ispisati `-1`.

Napomena: Zadatak mora biti rešen korišćenjem binarne pretrage i dinamičke alokacije memorije.

Primer 1

```
REZULTATI.TXT
Pera Petrovic 10
Mika Mikic 11
Zika Zikic 15.5
Marija Pavlovic 30.4
Marko Peric 31.6
Nebojsa Radic 41.2
Milutin Bojic 78.2
Andjela Jolic 79.2
STANDARDNI ULAZ:
78.2
STANDARDNI IZLAZ:
Milutin Bojic
```

Primer 2

```
REZULTATI.TXT
Pera Petrovic -10
Mika Mikic 11
Zika Zikic 15.5
Marija Pavlovic 30.4
Marko Peric 31.6
Nebojsa Radic 41.2
Milutin Bojic 78.2
Andjela Jolic 79.2
STANDARDNI IZLAZ ZA GREŠKE:
-1
```

Primer 3

```
Datoteka rezultati.txt
ne postoji
STANDARDNI IZLAZ ZA GREŠKE:
-1
```

Primer 4

```
REZULTATI.TXT
Marija Pavlovic 30.4
Marko Peric 31.6
Nebojsa Radic 41.2
Milutin Bojic 78.2
Andjela Jolic 79.2
STANDARDNI ULAZ:
100
STANDARDNI IZLAZ:
Niko nije ostvario
zadati broj poena
```