
Objektno programiranje (C++)

Prva domaća zadaća (ak. god. 2022./2023.)

Datum objave na Merlinu: četvrtak, 16. ožujka 2023. od 20:00h	Rok za predaju preko Merlina: nedjelja, 26. ožujka 2023. do 20:00h
Ukupan broj zadataka: 4 zadataka	Ukupno moguće ostvariti bodova: 10 bodova

Napomena. Na Merlinu je uz svaki zadatak bilo potrebno priložiti jednu .cpp datoteku s rješenjem tog zadatka, te u polje za unos teksta uz taj zadatak kratko opisati na koji način je taj zadatak bio riješen.

Zadatak 1. (3 boda) Napišite program koji od korisnika učitava tekst (do EOF). U tekstu se nalaze cijene iskazane u kunama. Cijene su iskazane na sljedeći način: nakon broja (s ili bez decimalne točke) nalazi se proizvoljno mnogo razmaka ili prelazaka u novi red, te zatim "HRK". Potrebno je ispisati uneseni tekst korisniku (na isti način kako je unesen), ali tako da su sve cijene u kunama iskazane u eurima (primjenjujemo tečaj konverzije 1 EUR = 7.5345 HRK; sve brojeve koji su dio cijene treba ispisati zaokružene na točno dvije decimale).

Osim primitivnih tipova podataka (int, float, ...), dozvoljeno je korištenje stringova (nije dozvoljeno korištenje polja, vektora, ...). Nije dozvoljeno korištenje regularnih izraza. Za cijene je pri zaokruživanju dovoljno „odbaciti višak znamenki” (npr. 1.329 može biti zaokružen na 1.32).

Primjer. Za ulaz:

```
HRK i EUR John je na putovanju Hrvatskom koristio kune (HRK).  
Jednu litru mlijeka platio je 7.54HRK, a 1 kg mrkve platio je 10  
HRK. Iako je usao u neželjeni minus od -5000 HRK,  
ugodno se proveo.
```

potrebno je ispisati:

```
HRK i EUR
```

```
John je na putovanju Hrvatskom koristio kune (HRK).  
Jednu litru mlijeka platio je 1.00EUR, a 1 kg mrkve platio je 1.32  
EUR. Iako je usao u neželjeni minus od -663.61 EUR,  
ugodno se proveo.
```

□

Zadatak 2. (2 boda) Napišite program koji od korisnika učitava cijele brojeve iz intervala $[0, 1000]$ (do EOF). Učitane brojeve potrebno je spremiti u vektor u sortiranom redoslijedu (od većeg prema manjem). Zatim treba korištenjem obratnih iteratora ispisati sve (međusobno disjunktne i neprazne) intervale $[a, b] \subseteq [0, 1000]$, $a, b \in \mathbb{Z}$, u kojima se ne nalazi niti jedan učitani broj. Ispis mora biti takav da je ispisan najmanji mogući broj takvih intervala (npr. umjesto $[1, 2]$ i $[2, 10]$ ispisujemo samo $[1, 10]$).

Primjer. Za ulaz:

```
7 51 10 6 8 10 100 50 9
```

potrebno je ispisati:

```
[0, 5]
[11, 49]
[52, 99]
[101, 1000]
```

□

Zadatak 3. (3 boda) Napišite program koji od korisnika učitava riječi (do EOF). Zatim je potrebno ispisati riječi u obrnutom poretku od unosa, te za svaku riječ ispisati koliko se puta ukupno pojavila u unosu (naravno, riječ koja se javila 5 puta bit će prema tome ispisana ukupno 5 puta zajedno s brojem 5). Za spremanje unesenih podataka potrebno je koristiti deque.

Primjer. Za ulaz:

```
Ako bi bilo biti onda
bi biti bilo tako.
```

potrebno je ispisati:

```
tako. 1
bilo 2
biti 2
bi 2
onda 1
biti 2
bilo 2
bi 2
Ako 1
```

□

Zadatak 4. (2 boda) Ovaj zadatak potrebno je riješiti korištenjem jednog od sljedećih adaptera: `stack`, `queue`, `priority_queue`.

Korisnik unosi liniju teksta koja predstavlja neki aritmetički izraz. Potrebno je ispisati samo jesu li u tom izrazu ispravno postavljene zagrade (ispišite poruku "ISPRAVNO" ili poruku "NIJE ISPRAVNO"). Tri su vrste zagrada: (), { } i [].

Ispravno postavljene zagrade znači:

- Otvorene zagrade moraju biti zatvorene istom vrstom zagrada.
- Otvorene zagrade moraju biti zatvorene u ispravnom redoslijedu. Također, nije moguće zatvoriti zagradu koja nije još otvorena.
- Istaknimo i da su u praznom stringu zagrade ispravno postavljene.

Primjeri.

1. primjer - ulaz:

$(2 + 3 - 4 * [5 - 3] + [4 / 2] - 1 + 3) * (4 + 2)$

1. primjer - izlaz:

ISPRAVNO

2. primjer - ulaz:

$(2 + \{3 - 4 * [5 - 3] + [4 / 2] - 1 + 2\} + 3) * (4 + 2)$

2. primjer - izlaz:

NIJE ISPRAVNO

□