

R smer, Programske paradigme 2025, JUN1

Na *Desktop-u* se nalazi arhiva **ppr_jun1_2025_ImePrezime_AlasNalog.zip** koju po raspakivanju treba preimenovati korišćenjem svojih podataka. Na primer, za studenta Jovana Marića čiji je broj indeksa 205/2022, ime direktorijuma je **ppr_jun1_2024_JovanaMarića**. Svaki zadatak sačuvati u odgovarajućem poddirektorijumu.

1. **[Python (20%)]** U datoteci `exchange.json` nalazi se spisak valuta i njihovih kurseva u odnosu na USD u json formatu. Na prvom nivou se nalaze se podaci o provajderu, valuti u odnosu na koju su ostali kursevi, datumu preuzimanja podataka, vreme koje je proteklo od poslednje izmene i najzad listu valuta, pri čemu svaka valuta sadrži informaciju o njenom kursu u odnosu na USD. Napisati program koji kao će vršiti konverzije između valuta. Nazivi valuta se prosleđuju kao argumenti komandne linije i to najpre valuta iz koje se konvertuje, pa valuta u koju se konvertuje. Ukoliko je broj argumenata komandne linije neodgovarajući ili neka od valuta ne postoji u listi, ispisati poruku i prekinuti program. Ukoliko je prosleđen dobar broj argumenata komandne linije, unosi se iznos koji treba konvertovati. Na standardnom izlazu ispisati konvertovani iznos.

Primer 1*

```
Poziv: python3 1.py USD GBP
Ulaz: 10
Izlaz:
7.4399999999999995
```

Primer 2*

```
Poziv: python3 1.py GBP USD
Ulaz: 7.44
Izlaz:
10
```

Primer 3*

```
Poziv: python3 1.py BSD BYD
Ulaz: 10
Izlaz:
20
```

Primer 4*

```
Poziv: python3 1.py USD NEPOSTOJECA
Ulaz: 10
Izlaz:
Uneli ste nepostojecu valutu!
```

Primer 5*

```
Poziv: python3 1.py
Izlaz:
Nedovoljno argumenata!
```

Primer 6*

```
Poziv: python3 1.py USD GBP ABC
Izlaz:
Previše argumenata!
```

2. **[Haskell (20%)]** Napisati Haskell biblioteku za rad sa karakterima:

- (a) Napisati funkcije `smanjiSlova :: String -> String` koja sva velika slova u prosleđenoj reči pretvara u mala i `uvecajSlova :: String -> String` koja sva mala slova pretvara u velika i vraća dobijeni rezultat.
- (b) Napisati funkcije `izdvojBrojeve :: String -> String`, koja iz prosleđenog stringa izdvaja cifre i vraća string koji se sastoji samo od tih cifara i `izbrisiBrojeve :: String -> String` koja iz stringa uklanja sve cifre i vraća izmenjenu nisku.
- (c) Napisati funkciju `obrisiSeparatore :: String -> String`, koja uklanja sve separatore iz prosleđene niske i vraća izmenjenu nisku.
- (d) Napisati funkciju `sifruj :: String -> Char -> Maybe String`, koja na osnovu prosleđenog karaktera modifikuje nisku na sledeće načine:
 - ukoliko je prosleđen karakter "s", sva velika slova u nisci se smanjuju,
 - ukoliko je prosleđen karakter "u", sva mala slova se uvećavaju,
 - ukoliko je prosleđen karakter "i", potrebno je izdvojiti cifre iz niske,
 - ukoliko je prosleđen karakter "b", potrebno je obrisati cifre iz niske,
 - ukoliko je prosleđen karakter "o" potrebno je ukloniti sve separatore iz niske
 - inače, vratiti `Nothing`.

Primer 1*

```
Poziv: ghci 2.hs
Ulaz: sifruj "SManJI FoNt" 's'
Izlaz:
Just "smanji font"
```

Primer 2*

```
Poziv: ghci 2.hs
Ulaz: sifruj "SManJI FoNt, 1a2b34!" 'u'
Izlaz:
Just "SMANJI FONT"
```

Primer 3*

```
Poziv: ghci 2.hs
Ulaz: sifruj " SManJI FoNt, 1a2b34!" 'o'
Izlaz:
Just "SManJIFoNt,1a2b34!"
```

Primer 4*

```
Poziv: ghci 2.hs
Ulaz: sifruj " SManJI FoNt, 1a2b34!" '3'
Izlaz:
Just "1234"
```

Primer 5*

```
Poziv: ghci 2.hs
Ulaz: sifruj " SManJI FoNt, 1a2b34!" 'b'
Izlaz:
Just " SManJI FoNt, ab! "
```

Primer 6*

```
Poziv: ghci 2.hs
Ulaz: sifruj " SManJI FoNt, 1a2b34!" 'n'
Izlaz:
Nothing
```

3. **[Haskell (25%)]** Napisati biblioteku za pomoć studentskoj službi u arhiviranju i vođenju evidencije o studentima. Definirati tipove **StepenStudija** (osnovne, master, doktorske), **Student** (karakterisan brojem indeksa, imenom, prezimenom i stepenom studija) i **Rezultat** (svaki student ima opcioni rezultat predstavljen brojem poena). Obezbediti da se student može ispisati na standardni izlaz i porediti po jednakosti sa ostalim studentima po broju indeksa. Pretpostaviti da su sve unete vrednosti validne.

Primer 1*

```
Poziv: ghci 3.hs
Ulaz: print ( MkStudent "123/22" "Pera" "Peric" OsnovneStudije)
IzLAZ:
123/22 : Pera Peric : Osnovne studije
```

Primer 2*

```
Poziv: ghci 3.hs
Ulaz: stud1 = MkStudent "123/22" "Pera" "Peric" OsnovneStudije
stud2 = stud1{indeks = "321/22"}
stud3 = stud2{indeks = "123/22"}
stud2 == stud1
IzLAZ:
False
```

Primer 3*

```
Poziv: ghci 3.hs
Ulaz: stud1 = MkStudent "123/22" "Pera" "Peric" OsnovneStudije
stud2 = stud1{indeks = "321/22"}
stud3 = stud2{indeks = "123/22"}
stud3 == stud1
IzLAZ:
True
```

Primer 4*

```
Poziv: ghci 3.hs
Ulaz:
stud1 = MkStudent "123/22" "Pera" "Peric" OsnovneStudije
rezultatIspita = (stud1, Nothing)
rezultatIspita
IzLAZ:
(123/22 : Pera Peric : Osnovne studije,Nothing)
```

4. **[Prolog (20%)]** Napisati program koji dodeljuje različite cifre različitim karakterima tako da izraz:
MANET + MATISSE + MIRO + MONET + RENOIR = ARTISTS bude zadovoljen.
Rezultat ispisati tako da se jasno vidi koja promenljiva je dobila koju vrednost.
5. **[Python (15%)]** Napisati program koji pronalazi sve vrednosti promenljivih X, Y, Z i W za koje važi da je $X \geq 2 * W$, $Z \geq Y + 3$ i $X + Y + 11 * Z - 11 * W \leq 100$ pri čemu promenljive pripadaju narednim domenima $X \in \{1, 2, 3, \dots, 10\}$, $Y \in \{1, 3, 5, 7, \dots, 51\}$, $Z \in \{10, 20, \dots, 100\}$ i $W \in \{1, 2, \dots, 100\}$. Ukoliko nema rešenja, ispisati poruku na standardni izlaz.