

Logičko i funkcijsko programiranje

I kolokvijum

Na *Desktop*-u u direktorijumu *Rad* kreirati direktorijum *ImePrezime_BrIndeksa* i unutar njega sačuvati programe koji sadrže rešenja datih zadataka. Nazivi fajlova **moraju** biti oblika *Zadatak1.hs*, *Zadatak2.hs*, *Zadatak3.hs*, *Zadatak4a.hs* i *Zadatak4b.hs*. Ukoliko je predato rešenje u programskom jeziku *Scala*, ekstenzija odgovarajućeg rešenja treba da bude *scala*, dok je naziv objekta isti kao naziv ulaznog fajla.

- (Haskell/Scala) **(2 poena)** Definisati tip *Oblik* koji može imati dve vrednosti :
 - Ključna reč *Krug* iza koje sledi poluprečnik kruga (realan broj)
 - Ključna reč *Pravougaonik* iza koje slede dužina i širina pravougaonika (dva realna broja)Napisati funkciju *površina*, koja prihvata jedan *Oblik* i vraća površinu datog oblika.

```
Main> povrsina (Krug 2.0)
12.566371
```

```
Main> povrsina (Pravougaonik 2.0 3.0)
6.0
```

- (Haskell/Scala) **(5 poena)** Napisati funkciju *kasa* koja na osnovu predate funkcije cenovnik i liste koja sadrži naziv proizvoda i broj komada tog proizvoda, vraća ukupan iznos računa. Funkcija cenovnik prihvata naziv proizvoda (string) a vraća cenu tog proizvoda (ceo broj). Asortiman proizvoda je ograničen samo na proizvode koje su izlistani u Tabeli 1. Cene proizvoda su fiksne.

Tabela 1 Cenovnik

```
Main> kasa cenovnik [(„Mleko“,1), („Cokolada“,2)]
270
```

Proizvod	Cena
Mleko	80
Pavlaka	120
Cokolada	95
Hleb	45

- (Haskell) **(7 poena)** Napiši funkciju *najveceRastojanje* koja primi listu tačaka u ravni (predstavljene kao parovi koordinata), i vraća rastojanje između dve najudaljenije tačke.

```
Main> najveceRastojanje [(0, 0), (1, 1), (-2, 4), (5, -3), (-4, -4)]
5.830
```

- (Haskell) **(9 poena)** Igra *nastaviRec* se igra tako što prvi takmičar kaže reč, a zatim drugi takmičar kaže reč koja počinje istim slovom kojim se prethodna reč završila. Igra se igra sve dok jedan od takmičara ne uspe da nastavi niz. Napisati funkcije:
 - nastaviRec* koja za zadatu listu reči vraća dužinu najdužeg niza reči koji se može napraviti po pravilima igre *nastaviRec*.

```
Main> nastaviRec ["pismo", "avan", "funkcija", "azot", "nasip", "timok"]
4
```

- b) ***nastaviRecSpisak*** koja za zadatu listu reči vraća listu reči koje formiraju najduži niz reči koji se može napraviti po pravilima igre ***nastaviRec***

```
Main> nastaviRecSpisak ["pismo", "avan", "funkcija", "azot", "nasip",  
"timok"]  
["funkcija", "avan", "nasip", "pismo"]
```

Svaka reč se u listi javlja tačno jednom.

Korisne ugrađene funkcije :

- Prebacivanje stringa u broj - `read p :: Int`
- Glava liste - `head lista`
- Rep liste - `tail lista`
- Prebacivanje broja u string - `show n`
- Ispis stringa na ekran - `putStr string`