

PROGRAMIRANJE 3 – NALOGE Z IZPITOV

Osnove, generični tipi, izjeme, datoteke ...

Iz CSV v asociativni seznam (datoteke, izjeme)

Napiši metodo, ki za vhod sprejme imeni vhodne in izhodne datoteke. Vhodna datoteka vsebuje podatke v formatu CSV. Vsaka vrstica vsebuje en primer, lastnosti primerov pa so ločene z vejicami. Prva vrstica vsebuje imena lastnosti, prav tako ločena z vejicami. Metoda naj podatke iz vhodne datoteke prepíše v izhodno datoteko v obliki asociativnih seznamov – seznamov urejenih parov (lastnost, vrednost). Za pomoč glej spodnji primer. Metoda naj deluje za poljubno število primerov in lastnosti v vhodni datoteki. Če vhodne datoteke ni, sproži ustrezno izjemo. Prav tako sproži izjemo, če podatki na vhodni datoteki niso ustrezni. V sporočilu izjeme sporoči vrstico, v kateri se je pojavila napaka, in samo vhodno vrstico.

Vhodna datoteka

```
Ime, Starost, Teza
Marko, 28, 91
Neža, 25, 57
Vito, 35, 71
```

Izhodna datoteka

```
[(Ime, Marko), (Starost, 28), (Teza, 91)]
[(Ime, Neža), (Starost, 25), (Teza, 57)]
[(Ime, Vito), (Starost, 35), (Teza, 71)]
```

VrniNtiElement de Luxe (generični tipi, izjeme)

Sestavi metodo `VrniNtiElement`, ki sprejme tabelo generičnega tipa ter vrne njen n -ti (n je seveda parameter metode) element (če seveda obstaja, sicer pa naj sproži izjemo). Če je n negativen, naj vrne $|n|$ -ti element s konca. Primeri uporabe:

```
string[] blebetanje = {"bla", "blo", "bli"};
int[] tja = {10, 20, 30};
string niz1 = VrniNtiElement(blebetanje, 2); // dobimo "blo"
string niz2 = VrniNtiElement(blebetanje, -3); // dobimo "bla"
string niz3 = VrniNtiElement(blebetanje, -4); // sproži se
izjema
int tam = VrniNtiElement(tja, 1); // dobimo 10
int tu = VrniNtiElement(tja, 0); // sproži se izjema
int tuk = VrniNtiElement(tja, 4); // sproži se izjema
```

Pripravi tudi ustrezni testni program!

Brez trojk! (osnove)

Napiši metodo

```
void Krajsaj(int[] t)
```

ki iz vseh števil v dani tabeli celih števil pobriše vse števke 3. Tako npr. iz tabele [13,23,333,112,3] nastane [1,2,112]. Metodo preveri v programu.

Na datoteko brez oklepajev (datoteke)

Sestavi metodo

```
public static void BrezOklepajev(string imeVhod, string imeIzhod)
```

ki tekstovno datoteko, katere ime je v `imeVhod` prepíše na novo datoteko z imenom, kot ga določa `imeIzhod`. Pri tem naj iz vrstic, ki vsebujejo izraze v oklepajih, pobriše te izraze skupaj z oklepaji. V vsaki vrstici se lahko pojavi največ en par pravilno postavljenih oklepajev. V primeru, da vrstica ni ustrezno sestavljena, jo v izhodni datoteki nadomesti z vrstico "NAPAKA". Primer:

Vhodna datoteka	Izhodna datoteka
Peter (spet) zamuja.	Peter zamuja.
Jaz sem (napisana narobe	NAPAKA
To ni račun.	To ni račun.
2*2+(2-7)=8	2*2+=8
To)(ni prav.	NAPAKA
[(2)]	[]
Konec(.).	Konec.

Nasvet: napiši primerne pomožne funkcije.

Pripravi tudi ustrezeni testni program!

Števke so delitelji (osnove)

Napiši metodo, ki sprejme cela števila a , b in k ter poišče vsa cela števila iz intervala $[a, b]$, v katerih vsaka neničelna števka deli število k . Metoda naj vrne niz v eni od naslednjih dveh oblik.

Cela števila iz intervala $[10, 20]$, v katerih vsaka neničelna števka deli 666, so: 10, 11, 12, 13, 16, 20.

V intervalu $[20, 40]$ ni celih števil, v katerih vsaka neničelna števka deli 5.

Za šumnike, pravilno rabo predlogov `s/z` ter dvojine/množine ni treba skrbeti.

Pripravi tudi ustrezeni testni program!

Vive La Difference! (izjeme)

Ustvari program, ki uporabnika vpraša po dveh številih: zmanjševancu in odštevanecu. Program naj vrne njuno razliko (razlika = zmanjševanec - odštevanec). Obe števili morata biti nenegativni realni števili, poleg tega pa mora biti odštevanec manjši ali enak zmanjševancu. Če katerikoli od pogojev ni izpolnjen, naj program sproži izjemo (za vsako od treh napak svojo izjemo).

Zgledi:

```
>>>
Zmanjševanec: 10.2
Odštevanec: 7
3.1999999999999993
>>>
Zmanjševanec: 4
Odštevanec: 5
Odštevanec mora biti manjši od zmanjševanca!
>>>
Zmanjševanec: -3
Zmanjševanec mora biti nenegativno število!
>>>
Zmanjševanec: 0
Odštevanec: ab
Odštevanec mora biti nenegativno število!
>>>
```