

Programiranje 2 - pisni izpit

20. 6. 2023 ob 10.00 (čas pisanja: 90 minut)

Naloga 1 (20 točk)

Napiši metodi `String vDvojisko(int x)` in `int vDesetisko(String x)` za pretvorbo iz desetiškega v dvojiški in iz dvojiškega v desetiški številski sistem. Pri pisanju metod uporabi le osnovne javanske operatorje (in ne, na primer, vgrajene metode `toString()`, `parseInt()` in podobnih).

V razred `Naloga21` zapiši metodo `main()` z naslednjo vsebino:

```
public static void main(String[] args) {  
    int x = Integer.parseInt(args[0]);  
    System.out.println("x: " + x);  
    String dvojisko = vDvojisko(x);  
    System.out.println("b: " + dvojisko);  
    int y = vDesetisko(dvojisko);  
    System.out.println("d: " + y);  
}
```

Ob pravilno napisanih metodah bo ob klicu `java Naloga21 42` program izpisal

```
x: 42  
b: 101010  
d: 42
```

Pravilnost delovanja programa lahko preveriš s spletnim preverjevalnikom.

Naloga 2 (25 točk)

Za implementacijo množice, v kateri bomo hranili znake angleške abecede, bomo uporabili eno število tipa `int` in naslednje pravilo: i -ti znak angleške abecede ('a' je ničti, 'b' je prvi, 'c' je drugi, ..., 'z' je petindvajseti znak) je v množici x tedaj in le tedaj, ko je v x prižgan i -ti bit.

Naloga: napiši razred `Mnozica` z atributom `int elementi` (v njem boš shranjeval elemente množice) in napiši metode

```
boolean jeElement(char znak);           // vrne true, če je znak element množice
void dodajElement(char znak);           // dodaj element v množico
void brisiElement(char znak);           // odstrani element iz množice
String toString();                       // vrne seznam elementov množice

Mnozica unija(Mnozica m1, Mnozica m2)   // vrne unijo dveh množic
Mnozica presek(Mnozica m1, Mnozica m2)  // vrne presek dveh množic
```

Razred `Mnozica` zapiši v datoteko `Naloga22.java` (za izhodišče vzemi kar datoteko `Naloga22.java`, ki se nahaja na eUčilnici). Ob pravilno napisanem razredu `Mnozica` bo poganjanje metode `main()` razreda `Naloga22` izpisalo:

```
[a,b,d]
[b,d,y]
[a,b,d,y]
[b,d]
```

Naloga 3 (25 točk)

Napiši program `Naloga23` za izpis vsebine datoteke na način, kot jo izpiše program `HexDump`.

Natančneje: program naj odpre datoteko, katere ime je podano v prvem argumentu ob klicu programa, in izpiše vsak bajt njene vsebine na dva načina – kot šestnajstiško številko (na levi strani izpisa) in kot znak (na desni strani izpisa). V primeru, da bajt ne predstavlja znaka z ASCII kodo med 32 in 127, naj program na desni strani izpisa na mestu tega znaka izpiše piko. V vsaki vrstici izpisa naj bo prikazanih 16 bajtov.

Primer: Če datoteka `a.txt` vsebuje naslednjo vsebino:

```
Danes je lepo.  
Jutri bo dež.
```

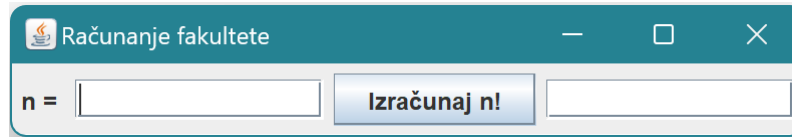
Naj program ob klicu `java Naloga23 a.txt` izpiše

```
44 61 6E 65 73 20 6A 65 20 6C 65 70 6F 2E 0D 0A    |Danes je lepo...|  
4A 75 74 72 69 20 62 6F 20 64 65 C5 BE 2E 0D 0A    |Jutri bo de.....|
```

Pravilnost delovanja programa lahko preveriš s spletnim preverjevalnikom.

4. Naloga (30 točk)

Napiši program, ki izriše okno z napisom, vpisnim poljem, gumbom in poljem za izpis, kot je prikazano na spodnji sliki.



Ob pritisku na gumb naj program prebere vrednost, ki je zapisana v levem vpisnem polju, izračuna fakulteto te vrednosti in rezultat zapiše v desno polje.

Dodatne zahteve:

- 1) v primeru, da je v levem vpisnem polju zapisana napačna vrednost (nekaj, kar ni število), naj program sproži nepreverljivo izjemo tipa `NapacnaVrednost`;
- 2) za podajanje poslušalca akcije pritiska na gumb uporabi anonimni notranji razred;
- 3) metoda za izračun fakultete naj bo rekurzivna!

Pomoč: poslušalec akcije pritiska na gumb je vsak razred, ki implementira vmesnik `ActionListener` in metodo `actionPerformed(ActionEvent e)`.