

Vježbe 4

1. Napisati JAVA program kojim se otvara .txt fajl i određuje i ispisuje statistika sadržaja tog fajla. Statistika sadrži:

- Broj rečenica fajla
- Broj riječi fajla
- Broj karaktera fajla
- Broj brojeva sadržanih u fajlu (npr. 3.45, -1.095678 ...)

```
import java.io.*;
public class Test {

    public static void main(String[] args) {
        String red;
        String[] rijeci;

        int brRecenica = 0;
        int brRijeci = 0;
        int brKaraktera = 0;
        int brBrojeva = 0;

        try {
            BufferedReader a = new BufferedReader(new FileReader(
                "/home/stefan/Desktop/tekst.txt"));

            while((red = a.readLine()) != null) {
                rijeci = red.split("\\s+")
                for(String s : rijeci) {
                    brRijeci++;
                    brKaraktera = brKaraktera + s.length();

                    if(s.matches(".*\\.|.*\\?|..*\\!")
                        brRecenica++;

                    if(s.matches("[+-]?([0|[1-9]\\d*)(\\.\\d+)?"))
                        brBrojeva++;
                }
            }
            a.close();

        } catch(IOException e) {
            e.printStackTrace();
        }

        System.out.println("Recenica:\n" + brRecenica + "\n");
        System.out.println("Rijeci:\n" + brRijeci + "\n");
        System.out.println("Karaktera:\n" + brKaraktera + "\n");
        System.out.println("Brojeva:\n" + brBrojeva + "\n");
    }
}
```

Objašnjenje: Koristimo klasu FileReader da čitamo podatke iz fajl. Ova klasa pristupa direktno fajlu u fajlu sistemu (hard disku), što može biti sporo. Klasa BufferedReader je dodata kako bi se poboljšale performanse učitavanja, jer ona učitava fajl u RAM memoriju čime se značajno ubrzava pristup sadržaju fajla. Alternativa su statičke metode iz klase Files, koje smo koristili na prošlom času.

2. Napisati JAVA program koji od korisnika traži unos stringa i koji nakon toga provjerava da li uneseni string može predstavljati e-mail adresu. Pretpostaviti da je e-mail adresa oblika "aaa@aaa.cc". Sa "aaa" je označen niz karaktera koji se sastoji od slova, brojeva, tačaka i znakova '-' i '_'. Voditi računa da dvije tačke ("..") ne mogu biti jedna do druge. Sa "cc" je označen string koji se sastoji isključivo od dva ili tri slova. Smatrati da će uneseni string imati samo mala slova jer e-mail adrese nijesu case-sensitive. Uz znak '@' se ne može nalaziti tačka (nije dozvoljeno "aaa.@aaa" ili "aaa@.aaa").
Kako bi se regularni izraz promijenio ako bi string "cc" (domen) ograničili na neku od vrijednosti ["edu", "com", "org", "net"]?

```
import java.util.Scanner;
public class Test {

    public static void main(String[] args) {

        String provjera;
        Scanner un = new Scanner(System.in);

        while(true){
            System.out.println("Unesite string za provjeru i 0 ukoliko želite da
                                izadjete: \n");
            provjera = un.next();

            if(provjera.matches("0"))
                break;

            if (provjera.matches("[_a-z0-9-]+(\\.[_a-z0-9-]+)*@[a-z0-9-]+(\\.[a-z0-9-]+)*"
                                "(\\.[a-z]{2,3})"))
                System.out.println("Može biti e-mail adresa.");
            else
                System.out.println("Ne može biti e-mail adresa.");
        }
    }
}
```

Za ograničene vrijednost domena regularni izraz bi bio:

```
"[_a-z0-9-]+(\\.[_a-z0-9-]+)*@[a-z0-9-]+(\\.[a-z0-9-]+)*"
"(\\.(edu|org|net|com))"
```