

## Objektum orientált programozás

### 2. gyakorlat (Java)

#### Algoritmizálási feladatok, ellenőrzött beolvasás, kétdimenziós tömb, String típus

1. 10-től indulva írja ki az első N db relatív prím számpárt. N-et ellenőrzöten olvassa be; értéke 1 és 10 közé essen. Két szám egymáshoz képest relatív prím, ha legnagyobb közös osztójuk 1.

```
//egy int érték ellenőrzött beolvasása Scanner objektummal
Scanner input = new Scanner(System.in);
int n = 0;
do {
    System.out.println("Enter a number between 1 and 10");
    while (!input.hasNextInt()) {
        System.out.println("That's not a number");
        input.next();
    }
    n = input.nextInt();
} while (n<1 || n>10);

//egy int érték ellenőrzött beolvasása BufferedReader objektummal
InputStreamReader instream = new InputStreamReader(System.in);
BufferedReader in = new BufferedReader(instream);
String str;
int n = 0;
do {
    try {
        System.out.print("Enter a number: ");
        str = in.readLine();
        n = Integer.parseInt(str);
    } catch (Exception e) {
        System.out.println("Not a number.");
    }
} while (n<1 || n>10);
```

2. Írja ki a számokat az értéküknek megfelelő számszor egymás után! A feladatot címkézett ciklussal oldja meg. A sorok számát ellenőrzöten olvassa be, értéke 1 és 9 közé essen. Pl: n=5-re

```
1
2 2
3 3 3
4 4 4 4
5 5 5 5 5
```

3. Deklaráljon egy String tömböt. A méretét olvassa be ellenőrzött módon; értéke 1 és 10 között legyen. Olvasson be a tömbbe neveket.

a) Állapítsa meg, hány azonos név szerepel a tömbben. Két sztring azonos, ha a `str1.equals(str2)` hívás eredménye `true`. Az összehasonlítás kis- és nagybetű érzékeny.

b) Számolja meg, hány azonos kezdőbetűs névpár van a tömbben. Sztring egy karakterét a `str.charAt(pozíció)` hívással kapjuk. A karakterek az `==` operátorral összehasonlíthatók.

**Figyelem!** A `nextInt()` az Enter karaktert nem olvassa be (az input bufferben hagyja), ezért a következő olvasási művelet előtt ezt ki kell olvasni az input bufferből a `nextLine()` hívással. A probléma elkerülhető, ha az int beolvasást külön függvényben valósítja meg.

4. Deklaráljon és inicializáljon egy tetszőleges méretű kétdimenziós int tömböt és külön függvényben valósítsa meg a mátrix elemeinek kiírását és a lineáris keresés algoritmusát. A mátrix sorainak száma: *matrix.length* ; oszlopainak száma : *matrix[0].length*. A keresendő elemet külön függvényben, ellenőrzött módon olvassa be.

5. Egy String mátrixba olvasson be 5 nevet és a hozzá tartozó email címeket. Egy embernek lehet több e-mail címe, számukat kérje be ellenőrzött módon (0 és 3 között). Tehát most olyan kétdimenziós tömböt kell létrehozni, amelyben az oszlopok száma soronként különböző lehet (jagged array). Hány embernek van egynél több email címe?