

**Teme:**

- ponavljanja.
  - stavki:
    - while
    - do-while
    - for
- dokumentiranje izvirne kode & javadoc

**Opomba:**

Za naslednje 4 naloge se zahteva rešitev, kjer boste za ponavljanja izključno uporabljali stavek **while**.

**Naloga 1**

Napišite javanski program, ki vrednost N na zaslon izpiše N-krat. N naj bo pri tem vhodni argument programa. Primer izvedba programa s parametrom 6:

```
>java V4n1 6  
666666
```

**Naloga 2**

Napišite javanski program ki na zaslon 'izriše' kvadrat vrednosti N s stranico N. Primer pri vnosu vrednosti 5.:

```
>java V4n2 5  
55555  
55555  
55555  
55555  
55555
```

### Naloga 3

Napišite javanski program ki na zaslon izriše podano obliko. Za boljše ponazoritev sta podana dva izrisa, prvi daje vizualizacijo izpisa pri izvedbi z argumentom 5, drugi pri vnosu argumenta z vrednostjo 3:

```
>java V4n3 5
54321
 4321
   321
    21
```

```
>java V4n3 3
 321
   21
```

### Naloga 4

Program naj razstavi z argumentom podano število na posamezne številke. Vsako številko podanega števila naj izpiše v lastni vrstici. Npr.:

```
>java V4n4 2319
2
3
1
9
```

### Naloga 5

Predhodne (zgornje) 4 primere izvedite (rešite) z uporabo stavka oblike do-while.

### Naloga 6

Naloge od 1 do 4 rešite z uporabo stavka for.

Opomba:  
*Naslednji dve nalogi sta za sprostitev*

## Naloga 7

Ustvari besedilno (ASCII) datoteko z imenom **Aplikacija.tmpl** in s spodaj podano vsebino. (Da se ne bi preveč mučili s prepisovanjem, naredite kopijo datoteke **stdcalss.tmpl** iz mape ....\BlueJ\lib\english\templates\newclass in kopijo poimenujte z denim imenom. Naredite zahtevane spremembe). Datoteko prepišite v mapo ....\BlueJ\lib\english\templates\newclass. (ista ciljna mapa!)

Prekinite delovanje programa BlueJ, če je ta slučajno zagnan, in ga ponovno zaženite. Ustvarite nov project, znotraj njega nov razred. Na spisku predlog za kreiranje novega razreda bi se morala pojaviti nova predloga. Uporabite jo. Oglejte si vsebino generiranega razreda.

Uporabite **Tools->Project documentation**. Orodje bi moralo zgenerirati dokumentacijo o vašem razredu v isto mapo, kjer se nahaja project. Primerjajte generirano dokumentacijo z vsebino razreda in si razložite pomen posameznih označb v izvorni kodi razreda (označbe se pričenjajo z znakom @).

Vsebina za Aplikacija.tmpl :

```
$PKGLINE
/**
 * Tule spišite opis razreda $CLASSNAME .
 *
 * @author (tvoje ime)
 * @version (stevilka verzije ali datum)
 */
public class $CLASSNAME {

    /**
     * Glavna metoda aplikacije - spisite komentar po svoji meri
     *
     * @param arg[0] - primer parametra glavne funkcije
     */
    public static void main(String[] arg)
    {
        // tule spišite svojo kodo
        System.out.println(arg[0]);
    }
}
```

Od te naloge dalje, bodo vsi vaši programi / razredi, paketi, vmesniki / ustrezno komentirani vsaj v skladu s podanim.

## Naloga 8

Poskusite dokumentacijo zgenerirati z interaktivnim orodjem javadoc (v mapi bin javanske namestitve) ! V pomoč morebiti naslednja 3 izvajanja (da, v tistem črnem oknu ... ):

```
C:\> \BlueJ500\jdk-17.0.2\bin\javadoc [enter]
C:\> \BlueJ500\jdk-17.0.2\bin\javadoc --help [enter]
C:\mojiJavaProjekti\vaja7_1> \BlueJ500\jdk-17.0.2\bin\javadoc Vaja7_1_naloga1.java [enter]
```

Opomba:

*Pa spet nazaj na programiranje ...*

## Naloga 9

Tabelirajte funkcijo  $y=\sin(x)$  na intervalu od  $-\pi/4$  do  $+\pi/4$  v koraku  $1/10$ . Izpisana tabela naj bo oblike:

```
x | y
-----
|
```

Pri tem naj bo tabela vizualno lepo oblikovana in poravnana na zaslonu. Omejite necele vrednosti števil na največ tri mesta za decimalno vejico (piko). Metoda za izračun sinus se nahaja v razredu Math (vsebine tečaja).

## Naloga 10\*

Narišite funkcijo iz predhodne naloge na izbranem intervalu na zaslon (glej podporno datoteko). Meje izbranega intervala privzamete iz argumentov ukazne vrstice (vpišete pri zagonu programa). /\* V pomoč: 'rišete' v koordinatni sistem, zasukan za  $-90^\circ$  (v smer urinega kazalca), x teče vertikalno, izračunana vrednost (y) pa se izrisuje horizontalno. \*/

## Naloga 11

Realizirajte izpis, kot je dan na spodnjem primeru. Upoštevajte, da je dolžina izpisa (število vrstic izpisa) voden podatek programa. Dan je primer za vhodni parameter z vrednostjo 6.

```
0 1 2 3 4 5
0 1 2 3 4
0 1 2 3
0 1 2
0 1
0
```

(Skrajni rok za dokončanje in oddajo vaje je 14 od datuma objave dokumenta.)

## Naloga 12

Realizirajte izpis, kot je dan na spodnjem primeru. Upoštevajte, da je dolžina izpisa (število vrstic izpisa) voden podatek programa. Dan je primer za vhodni parameter z vrednostjo 6. /\* V pomoč: vrstica 3 (?) v izpisu je sestavljena iz 6-3 presledkov in treh (3) vrednosti 2,1,0 – ... \*/

```
5 4 3 2 1 0
  4 3 2 1 0
    3 2 1 0
      2 1 0
        1 0
          0
```

## Naloga 13

Realizirajte tudi (glej besedilo predhodne vaje):

```
          0
         1 0
        2 1 0
       3 2 1 0
      4 3 2 1 0
     5 4 3 2 1 0
```

## Naloga 14

Realizirajte tudi (glej besedilo predhodne vaje):

```
0
0 1
0 1 2
0 1 2 3
0 1 2 3 4
0 1 2 3 4 5
```

**Naloga 15**

'Sestavite' predhodne 4 vaje v rezultativni izpis kot:

```

      0 0
    1 0 0 1
  2 1 0 0 1 2
3 2 1 0 0 1 2 3
4 3 2 1 0 0 1 2 3 4
5 4 3 2 1 0 0 1 2 3 4 5
5 4 3 2 1 0 0 1 2 3 4 5
4 3 2 1 0 0 1 2 3 4
  3 2 1 0 0 1 2 3
    2 1 0 0 1 2
      1 0 0 1
        0 0

```

(slika je rezultat rabe)  
(vhodnega argumenta z  
(vrednostjo 6

**Naloga 16**

Izvedite modifikacijo predhodne naloge :

```

      0
    1 0 1
  2 1 0 1 2
3 2 1 0 1 2 3
4 3 2 1 0 1 2 3 4
5 4 3 2 1 0 1 2 3 4 5
4 3 2 1 0 1 2 3 4
  3 2 1 0 1 2 3
    2 1 0 1 2
      1 0 1
        0

```

**Naloga 17**

In še naslednjo modifikacijo :

```

      0
    1 1
  2 2
3 3 3
4 4 4 4
5 5 5 5
4 4 4 4
3 3 3
2 2
  1 1
    0

```

(Skraini rok za dokončanje in oddajo vaje je 14 od datuma objave dokumenta.)