

Uvod u objektno orijentirano programiranje

Testiranje i razvoj

Debugging

Praktični primjer: Program za definiranje trokuta

```
public class Main {
    public static void main(String[] args) {
        Trokut t = new Trokut(1, 2, 3);
        Trokut t2 = new Trokut(1, 2, 3);
        Trokut[] polje = new Trokut[5];
        double c = t.bc + t.ab;
        System.out.println(t.ab + " " + t.bc + t.ac);
    }
}

public class Trokut {
    double ab, ac, bc;

    Trokut(double i, double j, double k) {
        ab = i;
        bc = j;
        ac = k;
    }
}
```

Debugging

Debug x

- Main (3) [Java Application]
 - Main at localhost:64152
 - Thread [main] (Suspended)
 - Trokut.<init>(double, double, double) line: 8
 - Main.main(String[]) line: 5

C:\Program Files\Java\jdk1.8.0_144\bin\javaw.exe (Dec 2, 2019, 1:52:41 PM)

Variables x **Expressions**

Name
no method return value
this
i
j
k

Navigator x

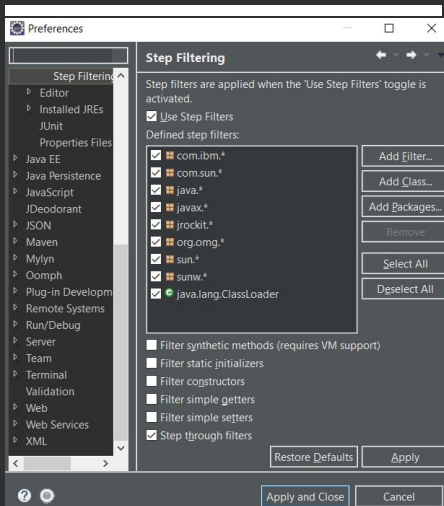
- Olimprijada
 - Vjezba1_zad1
 - Vjezba2
 - Vjezba3
 - Vjezba4
 - Vjezba6
- .settings
- bin
- src

Main.java

Trokut.java x **TrokutTest.java** **Main.java**

```
2 public class Trokut {
3
4
5     double ab,ac,bc;
6
7     Trokut(double i, double j, double k){
8         ab=i;
9         bc=j;
10        ac=k;
11    }
12 }
```

Debugging



Window

→ *Preferences*
→ *Java*
→ *StepFiltering*

- Javin programski okvir za testiranje programskih jedinica
- Testiranje metoda ili manjih dijelova određene funkcionalnosti
- Svaki put kada se kod izmjeni, test se i dalje izvodi s očekivanim rezultatom
- Kako bi definirali da se radi o testu, koristimo anotaciju @Test
- *RunAs* \rightarrow *JUnitTest*

Primjer

Želimo napraviti program koji provjerava da li je trokut definiran s tri dužine jednakostraničan, jednakokračan ili raznostraničan.

Potrebni testni slučajevi:

- Nije trokut
- Raznostraničan trokut
- jednakostraničan trokut
- Jednakokračan trokut

Zadatak

Implementirati zadatke i napisati testne slučajeve

Niven brojevi

Niven brojevi su brojevi koji su djeljivi sa zbrojem svojih znamenaka bez ostatka.

Sretni brojevi

Sretan broj je definiran slijedećim procesom:

- U pozitivnom integeru zamjeni broj sa sumom kvadrata znamenaka
- Proces se ponavlja sve dok broj nije jednak 1 ili se ponavlja zauvijek
- Ukoliko proces završi na 1, broj je sretan a u suprotnom nije :(
- Primjer sretnog broja:

$7 \rightarrow 49 \rightarrow 16 + 81 \rightarrow 97 \rightarrow 81 + 49 \rightarrow 130 \rightarrow 1 + 9 \rightarrow 10 \rightarrow 1$