


```

61     }
62
63     Subtraction subtract = new Subtraction(len, numbers);
64     subtract.subtraction();
65
66     // decider vraćamo na 0 kako bi nam se ponovo pojavio meni
67     decider = 0;
68 }
69
70 // Multiplication
71 if (decider == 3) {
72     System.out.println("-- Množenje --");
73     int len;
74
75     System.out.println("Koliko brojeva želite pomnožiti?" +
76         "(Brojevi će se množiti redosljedom kojim ih upisujete.)");
77     len = sc.nextInt();
78
79     Double[] numbers = new Double[len];
80     for (int counter = 0; counter < numbers.length; counter++) {
81         System.out.println("\t" + (counter+1) + ". broj: ");
82         numbers[counter] = sc.nextDouble();
83     }
84
85     Multiplication multiply = new Multiplication(len, numbers);
86     multiply.multiplication();
87
88     // decider vraćamo na 0 kako bi nam se ponovo pojavio meni
89     decider = 0;
90 }
91
92 // Dividing
93 if (decider == 4) {
94     System.out.println("-- Dijeljenje --");
95     int len;
96
97     System.out.println("Koliko brojeva želite podijeliti?" +
98         "(Brojevi će se dijeliti redosljedom kojim ih upisujete.)");
99     len = sc.nextInt();
100
101     Double[] numbers = new Double[len];
102     for (int counter = 0; counter < numbers.length; counter++) {
103         System.out.println("\t" + (counter+1) + ". broj: ");
104         numbers[counter] = sc.nextDouble();
105     }
106
107     Division divide = new Division(len, numbers);
108     divide.division();
109
110     // decider vraćamo na 0 kako bi nam se ponovo pojavio meni
111     decider = 0;
112 }
113
114 // Exit
115 if (decider == 9) {
116     System.out.println("Doviđenja!");
117     // test = false nas izbacuje iz do...while petlje
118     test = false;
119     break;
120 }

```

```
121         } while (test);  
122     }  
123 }
```

```
1 public class Addition {
2     int len;
3     Double[] numbers = new Double[len];
4
5     public Addition(int len, Double[] numbers) {
6         this.len = len;
7         this.numbers = numbers;
8     }
9
10    public int getLen() {
11        return len;
12    }
13
14    public void setLen(int len) {
15        this.len = len;
16    }
17
18    public Double[] getNumbers() {
19        return numbers;
20    }
21
22    public void setNumbers(Double[] numbers) { this.numbers = numbers; }
23
24    public void addition () {
25        // ovdje suma može biti 0 jer zbrajamo i neće nastati problemi
26        Double sum = 0.0;
27        for (Double number : numbers) sum += number;
28
29        System.out.println("Rezultat Vašeg zbroja je " + sum);
30    }
31 }
32
```

```
1 public class Division {
2     int len;
3     Double[] numbers = new Double[len];
4
5     public Division(int len, Double[] numbers) {
6         this.len = len;
7         this.numbers = numbers;
8     }
9
10    public int getLen() {
11        return len;
12    }
13
14    public void setLen(int len) {
15        this.len = len;
16    }
17
18    public Double[] getNumbers() {
19        return numbers;
20    }
21
22    public void setNumbers(Double[] numbers) { this.numbers = numbers; }
23
24    public void division() {
25        // Ova metoda mora biti drukcija od zbrajanja
26        // ako sum = 0; onda program dijeli 0 sa prvim elementom iz numbers
27        // npr. sum = 0, numbers[0] = 10; (sum /= numbers[0]) == 0
28        // zato suma = numbers[0] kako bi mogli dijeliti drugi element sa prvim itd.
29        Double sum = numbers[0];
30        for (int number = 1; number < len; number++) {
31            sum /= numbers[number];
32        }
33
34        System.out.println("Rezultat Vašeg dijeljenja je " + sum);
35    }
36 }
37
```

```
1 public class Subtraction {
2     int len;
3     Double[] numbers = new Double[len];
4
5     public Subtraction(int len, Double[] numbers) {
6         this.len = len;
7         this.numbers = numbers;
8     }
9
10    public int getLen() {
11        return len;
12    }
13
14    public void setLen(int len) {
15        this.len = len;
16    }
17
18    public Double[] getNumbers() {
19        return numbers;
20    }
21
22    public void setNumbers(Double[] numbers) { this.numbers = numbers; }
23
24    public void subtraction() {
25        // Ova metoda mora biti drukcija od zbrajanja
26        // ako sum = 0 onda program oduzima 0 od prvog elementa iz numbers
27        // npr. sum = 0, numbers[0] = 10; (sum -= numbers[0]) == -10
28        // zato suma = numbers[0] kako bi mogli oduzeti drugi element od prvog itd.
29        Double sum = numbers[0];
30        for (int number = 1; number < len; number++) {
31            sum -= numbers[number];
32        }
33
34        System.out.println("Rezultat Vašeg oduzimanja je " + sum);
35    }
36 }
37
```

```
1 public class Multiplication {
2     int len;
3     Double[] numbers = new Double[len];
4
5     public Multiplication(int len, Double[] numbers) {
6         this.len = len;
7         this.numbers = numbers;
8     }
9
10    public int getLen() {
11        return len;
12    }
13
14    public void setLen(int len) {
15        this.len = len;
16    }
17
18    public Double[] getNumbers() {
19        return numbers;
20    }
21
22    public void setNumbers(Double[] numbers) { this.numbers = numbers; }
23
24    public void multiplication() {
25        // Ova metoda mora biti drukcija od zbrajanja
26        // ako sum = 0; onda program dijeli 0 sa prvim elementom iz numbers
27        // npr. sum = 0, numbers[0] = 10; (sum *= numbers[0]) == 0
28        // zato suma = numbers[0] kako bi mogli dijeliti drugi element sa prvim itd.
29        Double sum = numbers[0];
30        for (int number = 1; number < len; number++) {
31            sum *= numbers[number];
32        }
33
34        System.out.println("Rezultat Vašeg množenja je " + sum);
35    }
36 }
37
```

```
1 "C:\Program Files\BellSoft\LibericaJDK-19-Full\bin\java.exe" "-javaagent:C:\Program Files\
JetBrains\IntelliJ IDEA 2022.2.3\lib\idea_rt.jar=57787:C:\Program Files\JetBrains\IntelliJ
IDEA 2022.2.3\bin" -Dfile.encoding=UTF-8 -Dsun.stdout.encoding=UTF-8 -Dsun.stderr.encoding=
UTF-8 -classpath D:\Faks\SuvremeneTehnikeProgramiranja\Zadace\zadaca05.A\out\production\
DZ05A Main
2
3 -----KALKULATOR-----
4
5
6 Koju operaciju želite izvesti?
7     1. Zbrajanje
8     2. Oduzimanje
9     3. Množenje
10    4. Dijeljenje
11    9. Izlaz
12 1
13 -- Zbrajanje --
14 Koliko brojeva želite zbrojiti?(Brojevi će se zbrojiti redosljedom kojim ih upisujete.)
15 3
16     1. broj:
17 12
18     2. broj:
19 25
20     3. broj:
21 43
22 Rezultat Vašeg zbroja je 80.0
23
24 Koju operaciju želite izvesti?
25     1. Zbrajanje
26     2. Oduzimanje
27     3. Množenje
28     4. Dijeljenje
29     9. Izlaz
30 2
31 -- Oduzimanje --
32 S Koliko brojeva želite napraviti oduzimanje?(Brojevi će se oduzeti redosljedom kojim ih
upisujete.)
33 4
34     1. broj:
35 45
36     2. broj:
37 15
38     3. broj:
39 7
40     4. broj:
41 -32
42 Rezultat Vašeg oduzimanja je 55.0
43
44 Koju operaciju želite izvesti?
45     1. Zbrajanje
46     2. Oduzimanje
47     3. Množenje
48     4. Dijeljenje
49     9. Izlaz
50 3
51 -- Množenje --
52 Koliko brojeva želite pomnožiti?(Brojevi će se množiti redosljedom kojim ih upisujete.)
53 3
54     1. broj:
55 15,44
```



```
56      2. broj:
57 12
58      3. broj:
59 0,75
60 Rezultat Vašeg množenja je 138.96
61
62 Koju operaciju želite izvesti?
63      1. Zbrajanje
64      2. Oduzimanje
65      3. Množenje
66      4. Dijeljenje
67      9. Izlaz
68 4
69 -- Dijeljenje --
70 Koliko brojeva želite podijeliti?(Brojevi će se dijeliti redosljedom kojim ih upisujete.)
71 8
72      1. broj:
73 954
74      2. broj:
75 12
76      3. broj:
77 3
78      4. broj:
79 1,2
80      5. broj:
81 4,2
82      6. broj:
83 1,034
84      7. broj:
85 15
86      8. broj:
87 2
88 Rezultat Vašeg dijeljenja je 0.16950149928873331
89
90 Koju operaciju želite izvesti?
91      1. Zbrajanje
92      2. Oduzimanje
93      3. Množenje
94      4. Dijeljenje
95      9. Izlaz
96 9
97 Doviđenja!
98
99 Process finished with exit code 0
100
```

Multiplication

m

Multiplication(int, Double[])

f

numbers

Double[]

f

len

int

m

multiplication()

void

p

len

int

p

numbers

Double[]

Subtraction

m

Subtraction(int, Double[])

f

len

int

f

numbers

Double[]

m

subtraction()

void

p

len

int

p

numbers

Double[]

Division

m

Division(int, Double[])

f

numbers

Double[]

f

len

int

m

division()

void

p

len

int

p

numbers

Double[]

Addition

m

Addition(int, Double[])

f

len

int

f

numbers

Double[]

m

addition()

void

p

len

int

p

numbers

Double[]

Main

m

Main()

m

main(String[]) void