

Міністерство освіти і науки України  
Національний університет «Львівська політехніка»



**Лабораторна робота №7**

з курсу:

**“ОБ’ЄКТНО ОРІЄНТОВАНЕ ПРОГРАМУВАННЯ”**

**Виконав:**

ст. гр.

КН-110

Халус

Богдан

**Прийняв:**

Гасько Р.Т.

Львів – 2018 р.

## Лабораторна робота № 7

**Завдання:** Пройти 7 тиждень курсу на платформі Прометеус.

### Виконання роботи

GitHub link - [https://github.com/bhalus/OOP\\_Java/tree/master/lab\\_7](https://github.com/bhalus/OOP_Java/tree/master/lab_7)

```
Application.java  RPNParserException.java
1  import java.util.Deque;
2  import java.util.LinkedList;
3
4  public class Application {
5
6      public static double parse(String str) {
7          Deque<Double> deque = new LinkedList<>();
8          StringBuilder strBuilder = new StringBuilder("");
9
10         str = str.trim();
11
12         for (int i = 0, length = str.length(); i < length; i++)
13         {
14             char sign = str.charAt(i);
15
16             if ((sign >= '0' && sign <= '9') || sign == '.' || (sign == '-' && i != length - 1 && str.charAt(i + 1) != ' '))
17             {
18                 strBuilder.append(sign);
19             }
20
21             else if (sign == ' ')
22             {
23                 if (strBuilder.length() > 0)
24                 {
25                     deque.push(Double.valueOf(strBuilder.toString()));
26                     strBuilder = new StringBuilder("");
27                 }
28             }
29
30             else if (sign == '+' || sign == '-' || sign == '*' || sign == '/')
31             {
32                 Double first, second;
33                 try
34                 {
35                     second = deque.pop();
36                     first = deque.pop();
37                 }
38                 catch (Exception e)
39                 {
40                     throw new RPNParserException();
41                 }
42
43                 switch (sign)
44                 {
45                     case '+':
46                         deque.push(first + second);
47                         break;
48                     case '-':
49                         deque.push(first - second);
50                         break;
51                     case '*':
52                         deque.push(first * second);
53                         break;
54                     case '/':
55                         if (second == 0)
56                         {
57                             throw new ArithmeticException();
58                         }
59                         deque.push(first / second);
60                         break;
61                 }
62             }
63         }
64
65         if (deque.size() > 1)
66         {
67             throw new RPNParserException();
68         }
69
70         return deque.pop();
71     }
72
73     public static void main(String[] args) {
74         if (args == null || args.length != 1 || args[0] == null)
75         {
76             throw new RPNParserException();
77         }
78         System.out.println(" = " + Application.parse(args[0]));
79     }
80 }
```

```
Application.java  RPNParserException.java
44  switch (sign)
45  {
46      case '+':
47          deque.push(first + second);
48          break;
49      case '-':
50          deque.push(first - second);
51          break;
52      case '*':
53          deque.push(first * second);
54          break;
55      case '/':
56          if (second == 0)
57          {
58              throw new ArithmeticException();
59          }
60          deque.push(first / second);
61          break;
62      }
63  }
64  else
65  {
66      throw new RPNParserException();
67  }
68  }
69  if (deque.size() > 1)
70  {
71      throw new RPNParserException();
72  }
73  return deque.pop();
74  }
75
76  public static void main(String[] args) {
77      if (args == null || args.length != 1 || args[0] == null)
78      {
79          throw new RPNParserException();
80      }
81      System.out.println(" = " + Application.parse(args[0]));
82  }
83  }
```