

Министерство цифрового развития, связи и массовых коммуникаций
Российской Федерации
Ордена Трудового Красного Знамени федеральное государственное
бюджетное образовательное учреждение высшего образования
“Московский технический университет связи и информатики”

Кафедра: "Математическая кибернетика и информационные
технологии"

Лабораторная работа №4 по дисциплине “Структуры и алгоритмы
обработки данных” по теме “Реализация стека/дека”

Выполнил студент
группы БФИ1902
вариант №9
Крутиков С.С.

г. Москва, 2021

Содержание

| | | |
|--------------|---|-----------|
| 1 | Задание на лабораторную работу | 3 |
| 2 | Ход выполнения лабораторной работы | 5 |
| 2.1 | Листинги программы | 5 |
| 2.2 | Результаты выполнения программы..... | 15 |
| Вывод | | 23 |

1 Задание на лабораторную работу

Задание на лабораторную работу представлено на рисунке 1.

1. Отсортировать строки файла, содержащие названия книг, в алфавитном порядке с использованием двух *деков*.
2. *Дек* содержит последовательность символов для шифровки сообщений. Дан текстовый файл, содержащий зашифрованное сообщение. Пользуясь *деком*, расшифровать текст. Известно, что при шифровке каждый символ сообщения заменялся следующим за ним в *деке* по часовой стрелке через один.
3. Даны три стержня и n дисков различного размера. Диски можно надевать на стержни, образуя из них башни. Перенести n дисков со стержня A на стержень C , сохранив их первоначальный порядок. При переносе дисков необходимо соблюдать следующие правила:
 - на каждом шаге со стержня на стержень переносить только один диск;
 - диск нельзя помещать на диск меньшего размера;
 - для промежуточного хранения можно использовать стержень B .
 Реализовать алгоритм, используя три *стека* вместо стержней A, B, C . Информация о дисках хранится в исходном файле.
4. Дан текстовый файл с программой на алгоритмическом языке. За один просмотр файла проверить баланс круглых скобок в тексте, используя *стек*.
5. Дан текстовый файл с программой на алгоритмическом языке. За один просмотр файла проверить баланс квадратных скобок в тексте, используя *дек*.
6. Дан файл из символов. Используя *стек*, за один просмотр файла напечатать сначала все цифры, затем все буквы, и, наконец, все остальные символы, сохраняя исходный порядок в каждой группе символов.
7. Дан файл из целых чисел. Используя *дек*, за один просмотр файла напечатать сначала все отрицательные числа, затем все положительные числа, сохраняя исходный порядок в каждой группе.
8. Дан текстовый файл. Используя *стек*, сформировать новый текстовый файл, содержащий строки исходного файла, записанные в обратном порядке: первая строка становится последней, вторая – предпоследней и т.д.
9. Дан текстовый файл. Используя *стек*, вычислить значение логического выражения, записанного в текстовом файле в следующей форме:

$$\langle \text{ЛВ} \rangle ::= \mathbf{T} \mid \mathbf{F} \mid (\mathbf{N} \langle \text{ЛВ} \rangle) \mid (\langle \text{ЛВ} \rangle \mathbf{A} \langle \text{ЛВ} \rangle) \mid (\langle \text{ЛВ} \rangle \mathbf{X} \langle \text{ЛВ} \rangle) \mid (\langle \text{ЛВ} \rangle \mathbf{O} \langle \text{ЛВ} \rangle),$$
 где буквами обозначены логические константы и операции:
 \mathbf{T} – True, \mathbf{F} – False, \mathbf{N} – Not, \mathbf{A} – And, \mathbf{X} – Xor, \mathbf{O} – Or.
10. Дан текстовый файл. В текстовом файле записана формула следующего вида:

$$\langle \text{Формула} \rangle ::= \langle \text{Цифра} \rangle \mid \mathbf{M}(\langle \text{Формула} \rangle, \langle \text{Формула} \rangle) \mid \mathbf{N}(\langle \text{Формула} \rangle, \langle \text{Формула} \rangle)$$

$$\langle \text{Цифра} \rangle ::= 0 \mid 1 \mid 2 \mid 3 \mid 4 \mid 5 \mid 6 \mid 7 \mid 8 \mid 9$$
 где буквами обозначены функции:
 \mathbf{M} – определение максимума, \mathbf{N} – определение минимума.
 Используя *стек*, вычислить значение заданного выражения.
11. Дан текстовый файл. Используя *стек*, проверить, является ли содержимое текстового файла правильной записью формулы вида:

$$\langle \text{Формула} \rangle ::= \langle \text{Терм} \rangle \mid \langle \text{Терм} \rangle + \langle \text{Формула} \rangle \mid \langle \text{Терм} \rangle - \langle \text{Формула} \rangle$$

Рисунок 1 – Задание на лабораторную работу

2 Ход выполнения лабораторной работы

2.1 Листинги программы

На листинге 1 представлен код в классе Task1, который отвечает за выполнение задания 1.

Листинг 1 – Код в Task1

```
import java.io.*;
import java.util.*;

public class Task1 {

    public static void main(String[] args) throws IOException {
        Deque<String> dec = new LinkedList<>();
        Deque<String> answer = new LinkedList<>();

        File file = new
File(System.getProperty("user.dir")+"\\src\\Task1.txt");
        Scanner scanner = new Scanner(file);
        while (scanner.hasNext()) {
            dec.add(scanner.nextLine());
        }
        scanner.close();

        String min = dec.getFirst();
        int m = dec.size();

        for (int i = 0; i < m-1; i++) {
            Iterator iterator = dec.iterator();
            while (iterator.hasNext()) {
                String s = (String) iterator.next();
                if (s.compareTo(min) < 0)
                    min = s;
            }

            answer.addLast(min);
            dec.remove(min);
            min=dec.getFirst();
        }

        answer.addLast(min);
        System.out.println(answer);
        String temp = answer.getFirst();

        try (PrintWriter pw = new
PrintWriter(System.getProperty("user.dir")+"\\src\\Task1_ans.txt")) {
            for (int i=0;i<m;i++){
                Iterator iterator = answer.iterator();
                while (iterator.hasNext()) {
                    String s = (String) iterator.next();
                    pw.println(s + "\n");
                }
                answer.remove(temp);
            }
        } catch (IOException exc) {
            System.out.println(exc);
        }
    }
}
```

```
}  
}
```

На листинге 2 представлен код в классе Task2, который отвечает за выполнение задания 2.

Листинг 2 – Код в Task2

```
import java.io.*;  
import java.util.Deque;  
import java.util.LinkedList;  
import java.util.Scanner;  
  
public class Task2 {  
  
    private final static char[] DEK =  
{ 'a', 'б', 'в', 'г', 'д', 'е', 'ё', 'ж', 'з', 'и', 'й', 'к', 'л', 'м', 'н', 'о', 'п', 'р', 'с',  
'т', 'у', 'ф', 'х', 'ц', 'ч', 'ш', 'щ', 'ъ', 'ы', 'ь', 'э', 'ю', 'я' };  
  
    public static void main(String[] args) throws IOException {  
        BufferedReader reader = new BufferedReader(new FileReader(new  
File(System.getProperty("user.dir")+"\\src\\Task2_info.txt")));  
        BufferedWriter writer = new BufferedWriter(new FileWriter(new  
File(System.getProperty("user.dir")+"\\src\\Task2.txt")));  
  
        int i = 0;  
  
        while((i = reader.read()) != -1) {  
            char ch = (char) i;  
            writer.append(switchLetter(ch));  
            writer.flush();  
        }  
  
        reader.close();  
        writer.close();  
  
        File f = new  
File(System.getProperty("user.dir")+"\\src\\Task2_info.txt");  
        System.out.println("Зашифрованное сообщение:");  
        out(f);  
  
        f=new File(System.getProperty("user.dir")+"\\src\\Task2.txt");  
        System.out.println("Расшифрованное сообщение:");  
        out(f);  
    }  
  
    private static char switchLetter(char ch) {  
        char outchar = '0';  
  
        for(int i = 2; i < DEK.length; i++) {  
            char c = DEK[i];  
            if(Character.compare(c, ch) == 0) {  
                outchar = DEK[i-2];  
                break;  
            }  
        }  
  
        if(Character.compare(outchar, '0') == 0)  
            outchar = ch;  
        return outchar;  
    }  
}
```

```

        public static void out (File f) throws FileNotFoundException {
            File file = new File(String.valueOf(f));
            LinkedList<String> answer = new LinkedList<>();
            Scanner scanner = new Scanner(file);

            while (scanner.hasNext()) {
                answer.add(scanner.nextLine());
            }

            System.out.println(answer);
        }
    }
}

```

На листинге 3 представлен код в классе Task3, который отвечает за выполнение задания 3.

Листинг 3 – Код в Task3

```

import java.io.*;
import java.util.Deque;
import java.util.LinkedList;
import java.util.Scanner;
import java.util.Stack;

public class Task3 {
    public static void main(String[] args) {
        Stack a = new Stack();
        Stack b = new Stack();
        Stack c = new Stack();

        System.out.println("Введите n:");
        Scanner scanner = new Scanner(System.in);
        int n = scanner.nextInt();

        for (int i=n;i>0;i--){
            a.add(i);
        }

        swap(a,b,c,n);

        System.out.println("Ответ:");
        System.out.println(c);
    }
    public static void swap (Stack a, Stack b, Stack c,int n){
        if (n == 1) {
            c.add(a.pop());
        }
        else {
            swap(a,c,b,n-1);
            c.add(a.pop());
            swap(b,a,c,n-1);
        }
    }
}

```

На листинге 4 представлен код в классе Task4, который отвечает за выполнение задания 4.

Листинг 4 – Код в Task4

```

import java.io.File;
import java.io.FileNotFoundException;
import java.util.*;

public class Task4 {

    public static void main(String[] args) throws FileNotFoundException {
        File file = new
File(System.getProperty("user.dir")+"\\src\\Task4.txt");
        Scanner scanner = new Scanner(file);

        Stack temp = new Stack();
        while (scanner.hasNext()) {
            temp.add(scanner.nextLine());
        }

        Iterator iterator = temp.iterator();

        int open=0, close=0;

        while (iterator.hasNext()){
            String s = (String) iterator.next();
            char[] ch= s.toCharArray();

            for (int i=0;i<s.length();i++){
                char o = '(';
                char c = ')';

                if (Character.compare(ch[i],o)==0) {
                    open++;
                }
                if (Character.compare(ch[i],c)==0) {
                    close++;
                }
            }
        }

        if (open == close)
            System.out.println("Количество символов совпадает");
        else
            System.out.println("Количество символов не совпадает");
    }
}

```

На листинге 5 представлен код в классе Task5, который отвечает за выполнение задания 5.

Листинг 5 – Код в Task5

```

import java.io.File;
import java.io.FileNotFoundException;
import java.util.*;

public class Task5 {

    public static void main(String[] args) throws FileNotFoundException {
        File file = new
File(System.getProperty("user.dir")+"\\src\\Task5.txt");
        Scanner scanner = new Scanner(file);

        Deque<String> temp = new LinkedList<String>();
    }
}

```



```

while (scanner.hasNext()) {
    temp.add(scanner.nextLine());
}

Iterator iterator = temp.iterator();

int open=0, close=0;

while (iterator.hasNext()){
    String s = (String) iterator.next();
    char[] ch= s.toCharArray();

    for (int i=0;i<s.length();i++){
        char o = '[';
        char c = ']';

        if (Character.compare(ch[i],o)==0) {
            open++;
        }

        if (Character.compare(ch[i],c)==0) {
            close++;
        }
    }
    if (open == close)
        System.out.println("Количество символов совпадает");
    else
        System.out.println("Количество символов не совпадает");
}
}

```

На листинге 6 представлен код в классе Task6, который отвечает за выполнение задания 6.

Листинг 6 – Код в Task6

```

import java.io.*;
import java.util.*;

public class Task6 {

    public static void main(String[] args) throws IOException {
        BufferedReader reader = new BufferedReader(new FileReader(new
        File(System.getProperty("user.dir")+"\\src\\Task6.txt")));

        Stack st = new Stack();
        Stack num = new Stack();
        Stack letter = new Stack();
        Stack symbols = new Stack();

        int i = 0;

        while ((i = reader.read()) != -1) {
            char ch = (char) i;
            st.add(ch);
        }

        Iterator iterator = st.iterator();

        while (iterator.hasNext()) {
            char r=(char) iterator.next();

```

```

        if (Character.isDigit(r)) {
            num.push(r);
        }
    }

    Iterator iterator1 = st.iterator();

    while (iterator1.hasNext()) {
        char r=(char) iterator1.next();
        if (Character.isAlphabetic(r)) {
            letter.push(r);
        }
    }

    Iterator iterator2 = st.iterator();

    while (iterator2.hasNext()) {
        char r=(char) iterator2.next();
        if (!(Character.isDigit(r) || Character.isAlphabetic(r))) {
            symbols.push(r);
        }
    }

    System.out.println("Числа:");
    System.out.println(num);
    System.out.println("Буквы:");
    System.out.println(letter);
    System.out.println("Символы:");
    System.out.println(symbols);
    System.out.println("Изначальный стек:");
    System.out.println(st);
}
}

```

На листинге 7 представлен код в классе Task7, который отвечает за выполнение задания 7.

Листинг 7 – Код в Task7

```

import java.io.*;
import java.util.*;

public class Task7 {

    public static void main(String[] args) throws IOException {
        Deque temp = new ArrayDeque();
        Stack s = new Stack();

        File file = new
File(System.getProperty("user.dir")+"\\src\\Task7.txt");
        Scanner scanner = new Scanner(file);

        String str = scanner.nextLine();
        String [] str1 = str.split(" ");

        for (int i = 0; i < str1.length; i++) {
            s.push(Integer.valueOf(str1[i]));
        }

        Iterator iterator = s.iterator();
        Iterator iterator1 = s.iterator();
    }
}

```

```

        while (iterator1.hasNext()) {
            int r = (int) iterator1.next();

            if (r<0){
                temp.add(r);
            }
        }

        while (iterator.hasNext()) {
            int r = (int) iterator.next();

            if (r>0){
                temp.add(r);
            }
        }
        System.out.println(temp);
    }
}

```

На листинге 8 представлен код в классе Task8, который отвечает за выполнение задания 8.

Листинг 8 – Код в Task8

```

import java.io.*;
import java.util.*;
public class Task8 {
    public static void main(String[] args) throws FileNotFoundException {
        File file = new
File(System.getProperty("user.dir")+"\\src\\Task8.txt");
        Scanner scanner = new Scanner(file);

        Stack temp = new Stack();

        while (scanner.hasNext()) {
            temp.add(scanner.nextLine());
        }

        try (PrintWriter pw = new
PrintWriter(System.getProperty("user.dir")+"\\src\\Task8_ans.txt")) {
            for (int i=0;i<=temp.size();++i) {
                String s;
                s = (String) temp.pop();
                pw.println(s);
            }
            pw.println(temp.pop());

            temp.remove(temp);
        }
        catch (IOException exc) {
            System.out.println(exc);
        }
    }
}

```

На листинге 9 представлен код в классе Task9, который отвечает за выполнение задания 9.

Листинг 9 – Код в Task9

```

import java.io.*;
import java.util.Iterator;
import java.util.Stack;

public class Task9 {
    public static void main(String[] args) throws IOException {
        BufferedReader reader = new BufferedReader(new FileReader(new
File(System.getProperty("user.dir")+"\\src\\Task9.txt")));

        Stack st = new Stack();
        Stack num = new Stack();
        Stack letter = new Stack();
        Stack preanswer = new Stack();

        int i = 0;

        while ((i = reader.read()) != -1) {
            char ch = (char) i;
            st.add(ch);
        }

        Iterator iterator = st.iterator();

        while (iterator.hasNext()) {
            char r=(char) iterator.next();
            if(Character.isDigit(r)){
                if (r=='0')
                    num.push(false);
                else
                    num.push(true);
            }
        }

        Iterator iterator1 = st.iterator();

        while (iterator1.hasNext()) {
            char r=(char) iterator1.next();
            if(Character.isAlphabetic(r)){
                letter.push(r);
            }
        }

        while (preanswer.size()!=6) {
            char s =(char) letter.pop();

            switch (s) {
                case ('O'):
                    boolean num1 = (boolean) num.pop();
                    boolean num2 = (boolean) num.pop();
                    preanswer.add(num1 || num2);
                    break;
                case ('X'):
                    boolean num3 = (boolean) num.pop();
                    boolean num4 = (boolean) num.pop();
                    if (num3 != num4) {
                        preanswer.add(true);
                    } else {
                        preanswer.add(false);
                    }
                    break;
                case ('A'):
                    boolean num5 = (boolean) num.pop();
                    boolean num6 = (boolean) num.pop();

```

```

        preanswer.add(num5 && num6);
        break;
    case ('N'):
        boolean num7 = (boolean) num.pop();
        if (num7 == true) {
            preanswer.add(false);
        } else {
            preanswer.add(true);
        }
        break;
    case ('F'):
        preanswer.add(false);
        break;
    case ('T'):
        preanswer.add(true);
        break;
    }

}

Iterator iterator2 = preanswer.iterator();

boolean answer=false;

while (iterator2.hasNext()) {
    if ((boolean)iterator2.next() == true){
        answer = true;
    }
}

System.out.println("ЛВ = " + answer);
}
}

```

На листинге 10 представлен код в классе Task10, который отвечает за выполнение задания 10.

Листинг 10 – Код в Task10

```

import java.io.*;
import java.util.Iterator;
import java.util.Stack;
public class Task10 {
    public static void main(String[] args) throws IOException {
        BufferedReader reader = new BufferedReader(new FileReader(new
File(System.getProperty("user.dir")+"\\src\\Task10.txt")));

        Stack st = new Stack();
        Stack num = new Stack();
        Stack letter = new Stack();
        Stack answer = new Stack();

        int i = 0;

        while ((i = reader.read()) != -1) {
            char ch = (char) i;
            st.add(ch);
        }

        Iterator iterator = st.iterator();

        while (iterator.hasNext()) {
            char r=(char) iterator.next();

```

```

        if(Character.isDigit(r)){
            num.add(r);
        }
    }

    Iterator iterator2 = st.iterator();

    while (iterator2.hasNext()) {
        char r=(char) iterator2.next();
        if(Character.isAlphabetic(r)){
            letter.push(r);
        }
    }

    while (answer.size()!=1){
        char s =(char) letter.pop();

        switch (s){
            case ('M'):
                char num1 =(char) num.pop();
                char num2 =(char) num.pop();
                if (num1>num2)
                    answer.add(num1);
                else
                    answer.add(num2);
                break;
            case ('N'):
                char num3 = (char) num.pop();
                char num4 = (char) num.pop();
                if (num4>num3)
                    answer.add(num3);
                else
                    answer.add(num4);
                break;
            case ('F'):
                char num6 = (char) num.pop();
                answer.add(num6);
        }
    }
    System.out.println(answer);
}
}

```

На листинге 11 представлен код в классе Task11, который отвечает за выполнение задания 11.

Листинг 11 – Код в Task11

```

import java.io.*;
import java.util.Iterator;
import java.util.Stack;
public class Task11 {
    public static void main(String[] args) throws IOException {
        BufferedReader reader = new BufferedReader(new FileReader(new
File(System.getProperty("user.dir")+"\\src\\Task11.txt")));

        Stack st = new Stack();
        Stack letter = new Stack();
        Stack symbols = new Stack();

        int i = 0;

```

```

while ((i = reader.read()) != -1) {
    char ch = (char) i;
    st.add(ch);
}

Iterator iterator1 = st.iterator();

while (iterator1.hasNext()) {
    char r = (char) iterator1.next();
    if (Character.isAlphabetic(r)) {
        letter.push(r);
    }
}

Iterator iterator2 = st.iterator();

while (iterator2.hasNext()) {
    char r = (char) iterator2.next();
    if (!(Character.isDigit(r) || Character.isAlphabetic(r))) {
        symbols.push(r);
    }
}

System.out.println(letter);
System.out.println(symbols);

int kol = 0;

while (symbols.size() != 0) {
    char s = (char) symbols.pop();
    switch (s) {
        case '+':
        case '-':
            kol++;
            break;
    }
}

int w = 0;

while (letter.size() != 0) {
    char s = (char) letter.pop();
    switch (s) {
        case 'x':
        case 'y':
        case 'z':
            w++;
            break;
    }
}

if (w-1 == kol)
    System.out.println("Формула имеет правильный вид");
else
    System.out.println("Формула имеет не правильный вид");
}
}

```

2.2 Результаты выполнения программы

Для выполнения задания 1 из файла Task1.txt были взяты исходные данные, которые представлены на рисунке 2.

The Adventures of Huckleberry Finn by Mark Twain, United States, (1835-1910)
 The Aeneid by Virgil, Italy, (70-19 BC)
 Anna Karenina by Leo Tolstoy, Russia, (1828-1910)
 Beloved by Toni Morrison, United States, (b. 1931)
 Berlin Alexanderplatz by Alfred Döblin, Germany, (1878-1957)
 Blindness by José Saramago, Portugal, (1922-2010)
 The Book of Disquiet by Fernando Pessoa, Portugal, (1888-1935)
 The Book of Job, Israel. (600-400 BC)
 The Brothers Karamazov by Fyodor M Dostoyevsky, Russia, (1821-1881)
 Buddenbrooks by Thomas Mann, Germany, (1875-1955)
 Canterbury Tales by Geoffrey Chaucer, England, (1340-1400)
 The Castle by Franz Kafka, Bohemia, (1883-1924)
 Children of Gebelawi by Naguib Mahfouz, Egypt, (b. 1911)
 Collected Fictions by Jorge Luis Borges, Argentina, (1899-1986)
 A Doll's House by Henrik Ibsen, Norway (1828-1906)
 Complete Poems by Giacomo Leopardi, Italy, (1798-1837)
 The Complete Stories by Franz Kafka, Bohemia, (1883-1924)
 The Complete Tales by Edgar Allan Poe, United States, (1809-1849)
 Absalom, Absalom! by William Faulkner, United States, (1897-1962)
 Confessions of Zeno by Italo Svevo, Italy, (1861-1928)
 Crime and Punishment by Fyodor M Dostoyevsky, Russia, (1821-1881)
 Dead Souls by Nikolai Gogol, Russia, (1809-1852)
 The Death of Ivan Ilyich and Other Stories by Leo Tolstoy, Russia, (1828-1910)
 Decameron by Giovanni Boccaccio, Italy, (1313-1375)
 The Devil to Pay in the Backlands by João Guimarães Rosa, Brazil, (1880-1967)
 Diary of a Madman and Other Stories by Lu Xun, China, (1881-1936)
 The Divine Comedy by Dante Alighieri, Italy, (1265-1321)
 Don Quixote by Miguel de Cervantes Saavedra, Spain, (1547-1616)
 Essays by Michel de Montaigne, France, (1533-1592)
 Fairy Tales and Stories by Hans Christian Andersen, Denmark, (1805-1875)
 Faust by Johann Wolfgang von Goethe, Germany, (1749-1832)
 Gargantua and Pantagruel by François Rabelais, France, (1495-1553)
 Gilgamesh Mesopotamia, (c 1800 BC)
 The Golden Notebook by Doris Lessing, England, (b.1919)
 Great Expectations by Charles Dickens, England, (1812-1870)
 Gulliver's Travels by Jonathan Swift, Ireland, (1667-1745)
 Gypsy Ballads by Federico García Lorca, Spain, (1898-1936)
 Hamlet by William Shakespeare, England, (1564-1616)
 History by Elsa Morante, Italy, (1918-1985)
 Hunger by Knut Hamsun, Norway, (1859-1952)
 The Idiot by Fyodor M Dostoyevsky, Russia, (1821-1881)
 The Iliad by Homer, Greece, (c 700 BC)
 Independent People by Halldór K Laxness, Iceland, (1902-1998)
 Invisible Man by Ralph Ellison, United States, (1914-1994)
 Jacques the Fatalist and His Master by Denis Diderot, France, (1713-1784)
 Journey to the End of the Night by Louis-Ferdinand Celine, France, (1894-1961)
 King Lear by William Shakespeare, England, (1564-1616)
 Leaves of Grass by Walt Whitman, United States, (1819-1892)
 The Life and Opinions of Tristram Shandy by Laurence Sterne, Ireland, (1713-1768)
 Lolita by Vladimir Nabokov, Russia/United States, (1899-1977)
 Love in the Time of Cholera by Gabriel García Márquez, Colombia, (b. 1928)
 Madame Bovary by Gustave Flaubert, France, (1821-1880)
 The Magic Mountain by Thomas Mann, Germany, (1875-1955)

Рисунок 2 – Исходные данные для задания 1

В результате выполнения программы был получен результирующий файл Task1_ans.txt. Его содержание показано на рисунке 3.

A Doll's House by Henrik Ibsen, Norway (1828-1906)

A Sentimental Education by Gustave Flaubert, France, (1821-1880)

Absalom, Absalom! by William Faulkner, United States, (1897-1962)

Anna Karenina by Leo Tolstoy, Russia, (1828-1910)

Beloved by Toni Morrison, United States, (b. 1931)

Berlin Alexanderplatz by Alfred Doblin, Germany, (1878-1957)

Blindness by Jose Saramago, Portugal, (1922-2010)

Buddenbrooks by Thomas Mann, Germany, (1875-1955)

Canterbury Tales by Geoffrey Chaucer, England, (1340-1400)

Children of Gebelawi by Naguib Mahfouz, Egypt, (b. 1911)

Collected Fictions by Jorge Luis Borges, Argentina, (1899-1986)

Complete Poems by Giacomo Leopardi, Italy, (1798-1837)

Confessions of Zeno by Italo Svevo, Italy, (1861-1928)

Crime and Punishment by Fyodor M Dostoyevsky, Russia, (1821-1881)

Dead Souls by Nikolai Gogol, Russia, (1809-1852)

Decameron by Giovanni Boccaccio, Italy, (1313-1375)

Diary of a Madman and Other Stories by Lu Xun, China, (1881-1936)

Don Quixote by Miguel de Cervantes Saavedra, Spain, (1547-1616)

Essays by Michel de Montaigne, France, (1533-1592)

Fairy Tales and Stories by Hans Christian Andersen, Denmark, (1805-1875)

Faust by Johann Wolfgang von Goethe, Germany, (1749-1832)

Gargantua and Pantagruel by Francois Rabelais, France, (1495-1553)

Gilgamesh Mesopotamia, (c 1800 BC)

Great Expectations by Charles Dickens, England, (1812-1870)

Gulliver's Travels by Jonathan Swift, Ireland, (1667-1745)

Gypsy Ballads by Federico Garcia Lorca, Spain, (1898-1936)

Hamlet by William Shakespeare, England, (1564-1616)

Рисунок 3 – Результат выполнения задания 1

Для выполнения задания 2 из файла Task2_info.txt были взяты исходные данные, которые представлены на рисунке 5.

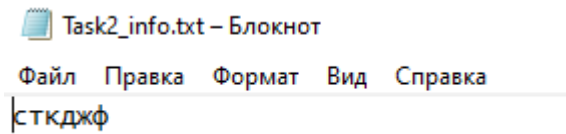


Рисунок 5 – Исходные данные для задания 2

В результате выполнения 2 был получен результирующий файл Task2.txt. Его содержание показано на рисунке 6.

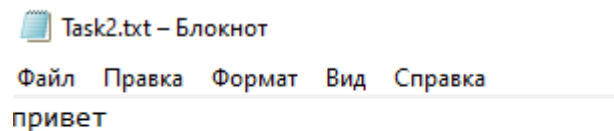


Рисунок 6 – Результат выполнения задания 2

В задании 3 пользователь должен ввести количество колец, которые будут надеты на первый стержень. Я ввел значение 4 и получил результат, который представлен на рисунке 7.



Рисунок 7 – Результат выполнения задания 3

Для выполнения задания 4 из файла Task4.txt были взяты исходные данные, которые представлены на рисунке 8.

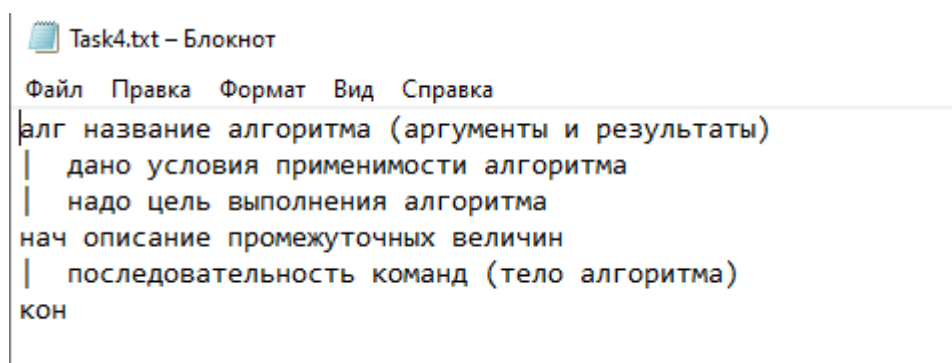


Рисунок 8 – Исходные данные для задания 4

В результате выполнения программы был получен результат, который представлен на рисунке 9.

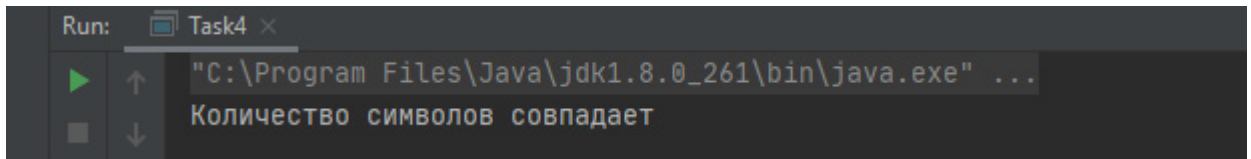


Рисунок 9 – Результат выполнения задания 4

Для выполнения задания 5 из файла Task5.txt были взяты исходные данные, которые представлены на рисунке 10.

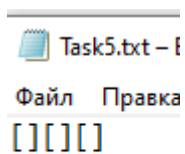


Рисунок 10 – Исходные данные для задания 5

В результате выполнения программы был получен результат, который представлен на рисунке 11.

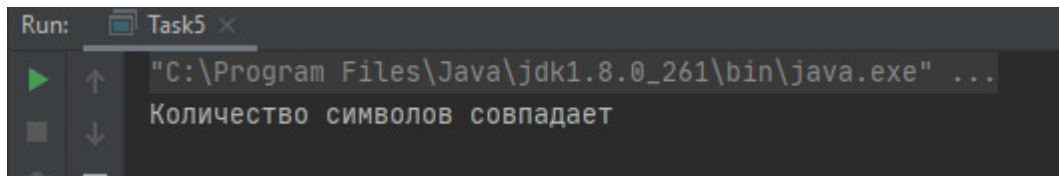


Рисунок 11 – Результат выполнения задания 5

Для задания 6 из файла Task6.txt были взяты исходные данные, которые представлены на рисунке 12.

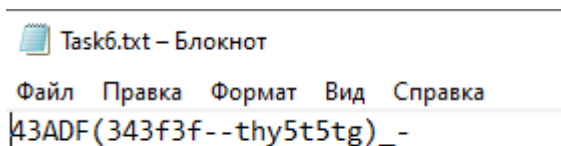


Рисунок 12 – Исходные данные для задания 6

В результате выполнения задания 6 был получен результат, который представлен на рисунке 13.

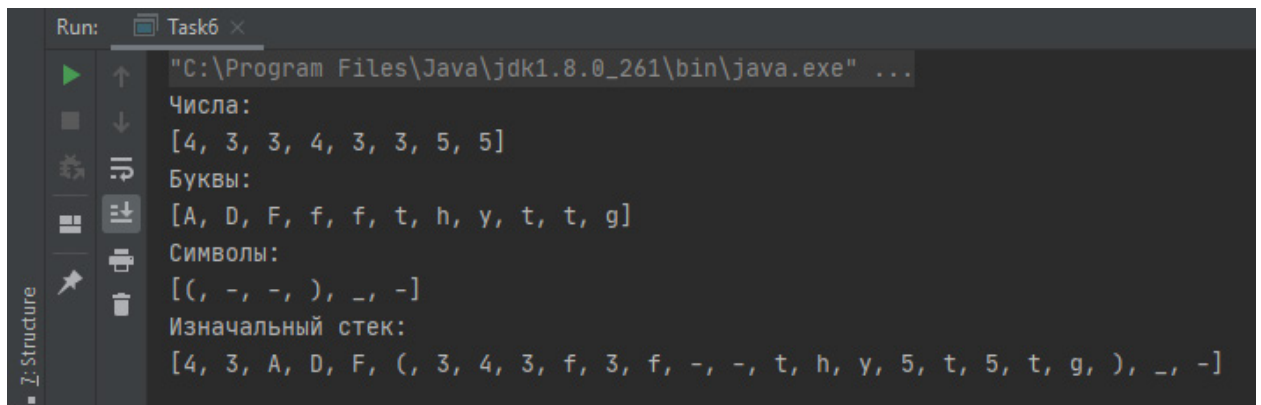


Рисунок 13 – Результат выполнения задания 6

Для задания 7 из файла Task7.txt были взяты исходные данные, которые представлены на рисунке 14.

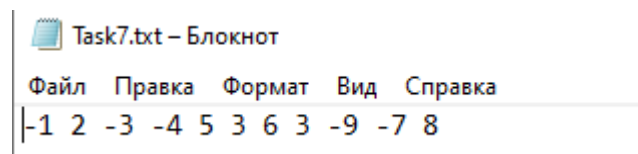


Рисунок 14 – Результат выполнения задания 7

В результате выполнения программы был получен результат, который представлен на рисунке 15.

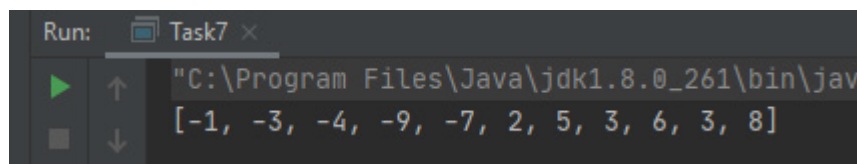


Рисунок 15 – Результат выполнения задания 7

Для задания 8 из файла Task8.txt были взяты исходные данные, которые представлены на рисунке 16.

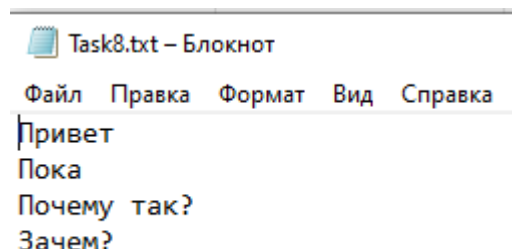


Рисунок 16 – Исходные данные для задания 8

Результат выполнения задания 8 записывается в файл Task8_ans.txt. Содержание данного файла после выполнения программы представлен на рисунке 17.

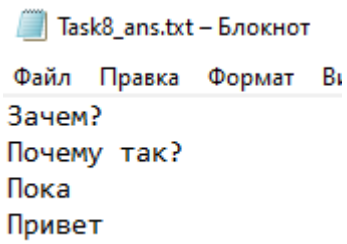


Рисунок 17 – Результат выполнения задания 8

Вывод

В результате выполнения данной лабораторной работы были получены навыки работы со стеком и деком.

Список использованных источников

- 1 ГОСТ 7.32-2017 Система стандартов по информации, библиотечному и издательскому делу. Отчет о научно-исследовательской работе. Структура и правила оформления
- 2 ГОСТ 7.1-2003 Библиографическая запись. Библиографическое описание. Общие требования и правила составления