

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 «Математической кибернетики и информационных технологий»

Лабораторная работа 4.

Реализация стека/дека.

Выполнил:

студент группы БВТ1902

Долматов Лев Евгеньевич

Москва
2021

Описание

Реализовать Стек, Дек, Разработать программу обработки данных, содержащихся в заранее подготовленном txt-файле, в соответствии с заданиями, применив указанную в задании структуру данных. Результат работы программы вывести на экран и сохранить в отдельном txt-файле. Выполнить 11 приложенных заданий.

Код

```
package com.company;

import javax.script.ScriptException;
import java.lang.reflect.Array;
import java.util.Collection;
import java.util.Deque;
import java.util.Iterator;
import java.util.LinkedList;

public class Main {

    public static void main(String[] args) throws ScriptException {
        //task1
        String str1="Aff Cfs Bwq De Dgjj SD ";
        new task1(str1);
        System.out.println();

        //task2
        String str = "12191(45197689)";
        Deque dec = new LinkedList();
        dec.addLast('1');
        dec.addLast('2');
        dec.addLast('T');
        dec.addLast('y');
        dec.addLast('9');
```

```

dec.addLast('(');
dec.addLast(' ');
dec.addLast('e');
dec.addLast('4');
dec.addLast('5');
dec.addLast('κ');
dec.addLast('c');
dec.addLast('6');
dec.addLast('7');
dec.addLast('π');
dec.addLast('л');
dec.addLast('8');
dec.addLast(')');
dec.addLast('я');
dec.addLast('3');
new task2(dec, str);
//task3
System.out.print("task3: ");
new task3(4, 'A','B','C');
//task4
String str4 = "(((1151()))())";
new task4(str4);
//task 5
String str5 = "[]";
new task5(str5);
//task6
String str6 = "111qqqwww111--1q-_+";
new task6(str6);
System.out.println();
//task7
int[] arr7 = {1, -15, 16, 71};
new task7(arr7);
System.out.println();
//task8
String str8 = "15 q1 1521 qwt";

```

```

        new task8(str8);
        System.out.println();
        //task9
        String str9="(F+(T*F+(F+T)))X(NF)";
        new task9(str9);
        //task10
        String str10="N(9,(M(1,N(1,2))))";
        new task10(str10);
        //task11
        String str11="x+(y+z+(z+y))";
        new task11(str11);
    }
}

package com.company;

import java.util.Deque;
import java.util.LinkedList;

public class task1 {
    public static boolean cpv(String str1, String str2) {
        boolean y1 = false;
        int len=Math.min(str1.length(),str2.length());
        for (int i = 0; i < len; i++) {
            if(str1.charAt(i)>str2.charAt(i)){
                return true;
            }
        }
        if(len==0)
            return true;
        return y1;
    }

    public static boolean cpv2(String str1, String str2) {
        boolean y1 = true;
        int len=Math.min(str1.length(),str2.length());

```

```

for (int i = 0; i < len; i++) {
    if(str1.charAt(i)!=str2.charAt(i)){
        return false;
    }
}
return y1;
}

```

```

public task1(String Str) {
    String[] words = Str.split(" ");
    Deque a = new LinkedList();
    Deque b = new LinkedList();
    for (int i = 0; i < words.length; i++) {
        a.push(words[i]);
    }
    String str = " ";
    while (!a.isEmpty()) {
        for (int i = 0; i < a.size(); i++) {
            if (cpv(str, a.peekFirst().toString())) {
                str = a.peekFirst().toString();
            }
            a.addLast(a.removeFirst());
        }
        for (int i = 0; i < a.size(); i++) {
            if (cpv2(str, a.peekFirst().toString())) {
                a.removeFirst();
                break;
            }
            a.addLast(a.removeFirst());
        }
        b.addLast(str);
        str="";
    }
    System.out.print("task1: ");
    while (!b.isEmpty()){

```

```

        System.out.print(b.removeFirst()+" ");
    }
}

package com.company;

import java.util.Deque;

public class task2 {
    public static boolean cpv(String str1, String str2) {
        boolean y1 = false;
        if (str1.charAt(0) == str2.charAt(0))
            y1 = true;
        return y1;
    }

    public task2(Deque dec, String str) {
        String str1 = "";
        for (int i = 0; i < str.length(); i++) {
            while (str1.length() < i + 1) {
                if (cpv(Character.toString(str.charAt(i)), dec.peekLast().toString())) {
                    dec.add(dec.pop());
                    dec.add(dec.pop());

                    str1 += dec.peekLast();
                }
                dec.add(dec.pop());
            }
        }
        System.out.println("task2: " + str1);
    }
}

package com.company;

```

```

import java.util.Deque;
import java.util.Stack;

public class task3 {

    public task3(int topN, char from,
                 char inter, char to) {
        if (topN == 1) {
            System.out.println("Disk 1 from "
                               + from + " to " + to);
        } else {
            new task3(topN - 1, from, to, inter);
            System.out.println("Disk "
                               + topN + " from " + from + " to " + to);
            new task3(topN - 1, inter, from, to);
        }
    }
}

package com.company;

```

```

import java.util.Deque;
import java.util.Stack;

public class task4 {
    public task4(String str) {
        Stack stc = new Stack();
        for(int i=0;i<str.length();i++){
            if(str.charAt(i)=='(')
                stc.push(str.charAt(i));
            else if(str.charAt(i)==')')
                stc.pop();
        }
        if(stc.empty())

```

```

        System.out.println("Task4: true");
    else
        System.out.println("Task4: false");
    }
}

package com.company;

import java.util.Deque;
import java.util.LinkedList;
import java.util.Stack;

public class task5 {
    public task5 (String str) {
        Deque stc = new LinkedList();
        for(int i=0;i<str.length();i++){
            if(str.charAt(i)=='[')
                stc.push(str.charAt(i));
            else if(str.charAt(i)==']'&&!stc.isEmpty())
                stc.remove();
        }
        if(stc.size()==0)
            System.out.println("Task5: true");
        else
            System.out.println("Task5: false");
    }
}

package com.company;

import java.util.Stack;

public class task6 {
    public task6(String str) {
        Stack num = new Stack();
        Stack sim = new Stack();
        Stack buk = new Stack();
    }
}

```



```

Stack per = new Stack();
for (int i = 0; i < str.length(); i++) {
    if (str.charAt(i) < '9' && '0' < str.charAt(i))
        num.push(str.charAt(i));
    else if ((str.charAt(i) < 'Z' && 'A' < str.charAt(i)) || (str.charAt(i) < 'z' && 'a' <
str.charAt(i)))
        buk.push(str.charAt(i));
    else
        sim.push(str.charAt(i));
}
System.out.print("task6: ");
while (!num.empty()) {
    per.push(num.pop());
}
while (!per.empty()) {
    System.out.print(per.pop());
} while (!buk.empty()) {
    per.push(buk.pop());
}
while (!per.empty()) {
    System.out.print(per.pop());
}
while (!sim.empty()) {
    per.push(sim.pop());
}
while (!per.empty()) {
    System.out.print(per.pop());
}

}
}

package com.company;

import java.lang.reflect.Array;
import java.util.Deque;

```

```

import java.util.LinkedList;
import java.util.Stack;

public class task7 {
    public task7(int[] arr) {
        Deque deq= new LinkedList();
        for (int i=0;i<arr.length;i++){
            if(arr[i]<0)
                deq.addLast(arr[i]);
            else
                deq.addFirst(arr[i]);
        }
        System.out.print("task7: ");
        for (int i=0;i<arr.length;i++){
            System.out.print(" "+deq.removeLast());
        }
    }
}

package com.company;

import java.util.Stack;

public class task8 {
    public task8(String str) {
        String[] words = str.split(" ");
        Stack stc = new Stack();
        System.out.print("task8: ");
        for (int i = words.length - 1; i >= 0; i--) {
            System.out.print(stc.push(words[i]) + " ");
        }
    }
}

package com.company;

import javax.script.ScriptEngine;

```

```

import javax.script.ScriptEngineManager;
import javax.script.ScriptException;
import java.util.Stack;

public class task9 {
    public static boolean cpv(String str1,String str2){
        boolean y1=false;
        if(str1.charAt(0)==str2.charAt(0))
            y1=true;
        return y1;
    }

    public task9(String Str) throws ScriptException {
        ScriptEngineManager manager = new ScriptEngineManager();
        ScriptEngine engine = manager.getEngineByName("JavaScript");
        StringBuilder str1 = new StringBuilder();
        Stack stk= new Stack();
        Stack stk2 = new Stack();
        for (int i = 0; i < Str.length(); i++) {
            stk.add(Str.charAt(i));
        }
        for (int i = 0; i < Str.length(); i++) {
            stk2.add(stk.pop());
        }
        for (int i = 0; i < Str.length(); i++) {
            if (cpv(stk2.peek().toString(), "T"))
                str1.append("true ");
            if (cpv(stk2.peek().toString(), "F"))
                str1.append("false ");
            if (cpv(stk2.peek().toString(), "N"))
                str1.append("! ");
            if (cpv(stk2.peek().toString(), "A") || cpv(stk2.peek().toString(), "*"))
                str1.append("&& ");
            if (cpv(stk2.peek().toString(), "X"))
                str1.append("!= ");
        }
    }
}

```

```

        if (cpv(stk2.peek().toString(), "O") || cpv(stk2.peek().toString(), "+"))
            str1.append("|| ");
        if (cpv(stk2.peek().toString(), "("))
            str1.append("(" ");
        if (cpv(stk2.peek().toString(), ")"))
            str1.append(")");
        stk2.pop();
    }

    Object o =engine.eval(str1.toString());
    System.out.println("task9: "+o);
}

}

package com.company;

import javax.script.ScriptEngine;
import javax.script.ScriptEngineManager;
import javax.script.ScriptException;
import java.util.Stack;

public class task10 {
    public static boolean cpv(String str1, String str2) {
        boolean y1 = false;
        if (str1.charAt(0) == str2.charAt(0))
            y1 = true;
        return y1;
    }

    public task10(String Str) throws ScriptException {
        ScriptEngineManager manager = new ScriptEngineManager();
        ScriptEngine engine = manager.getEngineByName("JavaScript");
        //Object o =engine.eval(code);
        StringBuilder str1 = new StringBuilder();
        Stack stk = new Stack();
        Stack stk2 = new Stack();

```

```

for (int i = 0; i < Str.length(); i++) {
    stk.add(Str.charAt(i));
    //System.out.println(stk.peek());
}
for (int i = 0; i < Str.length(); i++) {
    stk2.add(stk.pop());
    //System.out.println(stk.peek());
}
for (int i = 0; i < Str.length(); i++) {
    if (cpv(stk2.peek().toString(), "0"))
        str1.append("0");
    if (cpv(stk2.peek().toString(), "1"))
        str1.append("1");
    if (cpv(stk2.peek().toString(), "2"))
        str1.append("2");
    if (cpv(stk2.peek().toString(), "3"))
        str1.append("3");
    if (cpv(stk2.peek().toString(), "4"))
        str1.append("4");
    if (cpv(stk2.peek().toString(), "5"))
        str1.append("5");
    if (cpv(stk2.peek().toString(), "6"))
        str1.append("6");
    if (cpv(stk2.peek().toString(), "7"))
        str1.append("7");
    if (cpv(stk2.peek().toString(), "8"))
        str1.append("8");
    if (cpv(stk2.peek().toString(), "9"))
        str1.append("9");
    if (cpv(stk2.peek().toString(), "M"))
        str1.append("Math.max");
    if (cpv(stk2.peek().toString(), "N"))
        str1.append("Math.min");
    if (cpv(stk2.peek().toString(), ","))
        str1.append(",");

```

```

        if (cpv(stk2.peek().toString(), "("))
            str1.append("(" ");
        if (cpv(stk2.peek().toString(), ")"))
            str1.append(")");
        stk2.pop();
    }

    Object o = engine.eval(str1.toString());
    System.out.println("task10: " + o);
}

}

package com.company;

import javax.script.ScriptEngine;
import javax.script.ScriptEngineManager;
import javax.script.ScriptException;
import java.util.Stack;

public class task11 {
    public static boolean cpv(String str1, String str2) {
        boolean y1 = false;
        if (str1.charAt(0) == str2.charAt(0))
            y1 = true;
        return y1;
    }

    public task11(String Str) throws ScriptException {
        ScriptEngineManager manager = new ScriptEngineManager();
        ScriptEngine engine = manager.getEngineByName("JavaScript");
        //Object o =engine.eval(code);
        StringBuilder str1 = new StringBuilder();
        Stack stk = new Stack();
        Stack stk2 = new Stack();
        for (int i = 0; i < Str.length(); i++) {
            stk.add(Str.charAt(i));

```

```

        //System.out.println(stk.peek());
    }
    for (int i = 0; i < Str.length(); i++) {
        stk2.add(stk.pop());
        //System.out.println(stk.peek());
    }
    for (int i = 0; i < Str.length(); i++) {
        if (cpv(stk2.peek().toString(), "x"))
            str1.append("1");
        if (cpv(stk2.peek().toString(), "y"))
            str1.append("2");
        if (cpv(stk2.peek().toString(), "z"))
            str1.append("3");
        if (cpv(stk2.peek().toString(), "("))
            str1.append("(");
        if (cpv(stk2.peek().toString(), ")"))
            str1.append(")");
        if (cpv(stk2.peek().toString(), "+"))
            str1.append("+");
        if (cpv(stk2.peek().toString(), "-"))
            str1.append("-");
        stk2.pop();
    }
    try {
        Object o = engine.eval(str1.toString());
        System.out.println("task11: true" );
    } catch (Throwable e) {
        System.out.println("task11: false" );
    }
}
}

```

Вывод

Выполнив данную лабораторную работу, я научился реализовывать Стек и Дек, решать с помощью этих структур различного типа задачи.