**Федеральное агентство связи**

**Государственное бюджетное образовательное учреждение**

**высшего образования**

**ордена Трудового Красного Знамени**

**«Московский технический университет связи и информатики»**

Кафедра

«Математическая кибернетика и информационные технологии»

**Лабораторная работа №4**

**по дисциплине «Структура и алгоритмы обработки данных»**

Выполнил студент

Группы БФИ1902

Леонов Н.Н.

Научный руководитель: Мкртчян Г.М.

Москва 2021

# Реализация стека/дека

Задания на лабораторную работу:

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 А на стержень С**,**

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 В**.** Реализовать алгоритм, используя три **стека** вместо стержней А**,** В**,** С**.** Информация о дисках хранится в исходном файле.

1. Дан текстовый файл с программой на алгоритмическом языке. За один просмотр файла проверить баланс круглых скобок в тексте, используя **стек.**
2. Дан текстовый файл с программой на алгоритмическом языке. За один просмотр файла проверить баланс квадратных скобок в тексте, используя **дек.**
3. 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.
4. 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.
5. 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.
6. Дан текстовый файл. Используя ***стек*,** вычислить значение логического выражения, записанного в текстовом файле в следующей форме:

< ЛВ > ::= **T** | **F** | (**N**<ЛВ>) | (<ЛВ>**A**<ЛВ>) | (<ЛВ>**X**<ЛВ>) | (<ЛВ>**O**<ЛВ>),

где буквами обозначены логические константы и операции:

**T** – True, **F** – False, **N** – Not, **A** – And, **X** – Xor, **O** – Or.

1. Дан текстовый файл. В текстовом файле записана формула следующего вида:

<Формула> ::= <Цифра> | **M**(<Формула>,<Формула>) |

**N**(Формула>,<Формула>)

< Цифра > ::= 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 где буквами обозначены функции:

**M** – определение максимума, **N** – определение минимума. Используя ***стек***, вычислить значение заданного выражения.

1. Дан текстовый файл. Используя ***стек***, проверить, является ли содержимое текстового файла правильной записью формулы вида:

< Формула > ::= < Терм > | < Терм > + < Формула > | < Терм > - < Формула >

< Терм > ::= < Имя > | (< Формула >)

< Имя > ::= **x** | **y** | **z**

Ниже представлен код лабораторной работе

import java.io.\*; import java.util.\*; public class First {

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

Deque<String> answer = new LinkedList<>(); File file = new

File("C:\\Users\\Игнатиks\\IdeaProjects\\4семестр\\Lab4\\Books.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("C:\\Users\\WithLove\\IdeaProjects\\Siaod\_4\\First.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);

}

}

}

import java.util.Scanner;

public class Second {

private final static char[] *DEK* =

{'а','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("C:\\Users\\Игнатиks\\IdeaProjects\\4семестр\\Lab4\\Info.txt"))); BufferedWriter writer = new BufferedWriter(new FileWriter(new

File("C:\\Users\\Игнатиks\\IdeaProjects\\4семестр\\Lab4\\Second.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("C:\\Users\\Игнатиks\\IdeaProjects\\4семестр\\Lab4\\Info.txt"); System.*out*.println("Зашифрованное сообщение:");

*out*(f);

System.*out*.println("Расшифрованное сообщение:"); f=new

File("C:\\Users\\Игнатиks\\IdeaProjects\\4семестр\\Lab4\\Second.txt");

*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);

}

}

import java.util.Scanner; import java.util.Stack;

public class Third {

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);

}

}

}

import java.io.File;

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

public class Fourth {

public static void main(String[] args) throws FileNotFoundException { File file = new

File("C:\\Users\\Игнатиks\\IdeaProjects\\4семестр\\Lab4\\Fourth.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");

}

}

import java.io.File;

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

public class Fifth {

public static void main(String[] args) throws FileNotFoundException { File file = new

File("C:\\Users\\Игнатиks\\IdeaProjects\\4семестр\\Lab4\\Fifth.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");

}

}

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

public class Sixth {

public static void main(String[] args) throws IOException { BufferedReader reader = new BufferedReader(new FileReader(new

File("C:\\Users\\Игнатиks\\IdeaProjects\\4семестр\\Lab4\\Sixth.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);

}

}

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

public class Seventh {

public static void main(String[] args) throws IOException {

Deque temp = new ArrayDeque(); Stack s = new Stack();

File file = new File("C:\\Users\\Игнатиks\\IdeaProjects\\4семестр\\Lab4\\Seventh.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);

}

}

import java.io.\*; import java.util.\*; public class Eighth {

public static void main(String[] args) throws FileNotFoundException { File file = new

File("C:\\Users\\Игнатиks\\IdeaProjects\\4семестр\\Lab4\\Eighth.txt"); Scanner scanner = new Scanner(file);

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

temp.add(scanner.nextLine());

}

try (PrintWriter pw = new PrintWriter("C:\\Users\\Игнатиks\\IdeaProjects\\4семестр\\Lab4\\Eighth\_answer

.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);

}

}

}

import java.io.\*;

import java.util.Iterator; import java.util.Stack; public class Tenth {

public static void main(String[] args) throws IOException { BufferedReader reader = new BufferedReader(new FileReader(new

File("C:\\Users\\Игнатиks\\IdeaProjects\\4семестр\\Lab4\\Tenth.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);

}

}

import java.io.\*;

import java.util.Iterator; import java.util.Stack; public class Eleventh {

public static void main(String[] args) throws IOException { BufferedReader reader = new BufferedReader(new FileReader(new

File("C:\\Users\\Игнатиks\\IdeaProjects\\4семестр\\Lab4\\Eleventh.txt"))); Stack st = new Stack();

Stack letter = new Stack(); Stack symbols = new Stack(); int open = 0, close = 0; 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);

}

}

int kol = 0;

while (symbols.size() != 0) { char s = (char) symbols.pop(); switch (s) {

case ('+'):

case ('-'):

kol++; break;

case ('('):

open++; break;

case (')'):

close++; 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) && (close == open)) System.*out*.println("Формула имеет правильный вид");

else

}

}

System.*out*.println("Формула имеет не правильный вид");

На рисунках 1 – 14 будет представлен результат работы программы



Рисунок 1 – Задание № 1



Рисунок 2 – Задание № 1



Рисунок 3 – Задание № 1

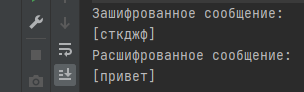


Рисунок 4 – Задание № 2

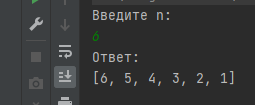


Рисунок 5 – Задание № 3



Рисунок 6 – Задание № 4



Рисунок 7 – Задание №5

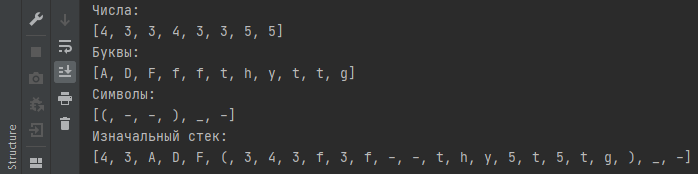


Рисунок 8 – Задание № 6



Рисунок 9 - Задание № 7

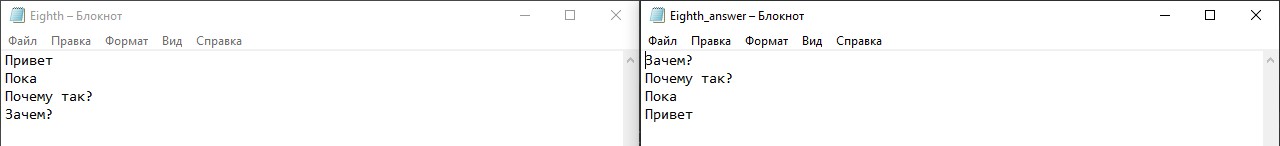


Рисунок 10 – Задание № 8



Рисунок 11 – Задание № 9

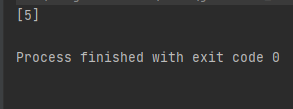


Рисунок 12 – Задание № 10

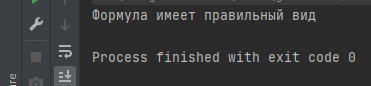


Рисунок 13 – Задание № 11