

|  |  |
| --- | --- |
| МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ  федеральное государственное бюджетное образовательное учреждение высшего образования  **«МОСКОВСКИЙ АВТОМОБИЛЬНО-ДОРОЖНЫЙ ГОСУДАРСТВЕННЫЙ**  **ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ (МАДИ)»** |  |

**Кафедра «Высшая математика»**

**Лабораторная работа № 1**

по дисциплине

«Структуры и алгоритмы обработки данных»

на тему:

**«**Реализация стека/дека**»**

**Выполнил:**

Учебная группа: 1бПМ   
ФИО: Греча К. П.

Подпись: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Руководитель лабораторной работы:**

Должность: старший преподаватель

Звание: б/з

ФИО: Кутейников И. А.

Подпись: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

«\_\_\_\_»\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_20\_\_г.

Цель:

Разработать программу обработки данных, содержащихся в заранее подготовленном файле, в соответствии с индивидуальным заданием. Программа должна включать модуль, содержащий набор всех необходимых средств (типов, подпрограмм и т.д.) для решения поставленной задачи.

Задание:

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.

Алгоритм:

import java.io.BufferedReader;  
import java.io.IOException;  
import java.io.FileReader;  
  
public class Main {  
  
 public static void main(String[] args) {  
 try {  
 BufferedReader reader = new BufferedReader(new FileReader("C:/Intellij Idea/txt.txt"));  
 String line = reader.readLine();// считывает строку и возвращает её  
  
 Stack digit = new Stack();  
 Stack letter = new Stack();  
 Stack other = new Stack();  
  
 while (line != null) {  
 for (char c : line.toCharArray()) { //смотрим каждый символ из символьного массива  
 if (Character.*isDigit*(c)) { //проверяем явл. ли он цифрой  
 digit.push(c);// если да, то цифра отправляется в стек  
 } else if (Character.*isLetter*(c)) { // тоже самое с буквами и с остальными символами  
 letter.push(c);  
 } else {  
 other.push(c);  
 }  
 }  
 line = reader.readLine(); // еще раз смотрим строку  
 }  
  
 System.*out*.println("Цифры:");  
 digit.printSt(); // Вывод стека  
 System.*out*.println("\nБуквы:");  
 letter.printSt();  
 System.*out*.println("\nОстальные символы:");  
 other.printSt();  
  
 reader.close(); //закрытие потока ввода ресурсов  
 } catch (IOException e) { // закрытия try и вывод ошибки ввода вывода  
 e.printStackTrace();  
 }  
 }  
  
 static class Node {  
 char data; //данные в узле  
 Node next;  
 public Node(char data) { // создание конструктора, а также инициализация узла  
 this.data = data;  
 }  
 }  
  
 static class Stack {  
 Node upper; // верхний элемент списка, последний добавленный  
 public boolean isEmpty() {  
 return upper == null; // проверка на пустоту стека  
 }  
  
 public void push(char data) {// добавление нового узла с символом дата в стек  
 Node newNode = new Node(data); //создание нового узла  
 newNode.next = upper; // создание ссылки узла на верхний элемент списка  
 upper = newNode;// новый узел становится верхним элементом списка  
 }  
  
 public char top() { // возвращает значение символа из стека но не удаляет его  
 if (isEmpty()) { // проверка на пустоту стека  
 throw new IllegalStateException("Стек пустой "); //Ошибка  
 }  
 char data = upper.data; // в дату записыв. значение из верхнего узла стека  
 upper = upper.next; // обновление ссылки  
 return data;  
 }  
  
 public void printSt() {  
 Stack reverseStack = new Stack(); // Создаем новый стек для обратного вывода  
 Node current = upper; // Устанавливаем текущий узел на верхний элемент стека  
 while (current != null) {  
 reverseStack.push(current.data); // Помещаем символы в новый стек на вершину стека  
 current = current.next; // Переходим к следующему узлу в стеке  
 }  
 while (!reverseStack.isEmpty()) { //цикл пока стек не пустой  
 System.*out*.print(reverseStack.top()+ " "); // Выводим символы в обратном порядке  
 }  
 }  
 }  
}

**Результат:**

**Данные в исходном файле:**

sd39jerd38u28 u18wd1jk\*#U\*MED#\*M@

#OMJDIM@<OK)SK)D#IDME@jdisjd92e8(@ID#\*IK(QSke9kdas;';]

Llpfdoe(odoi#k(d)w#289Q47919E E010 2KK

**Вывод:**

Цифры:

3 9 3 8 2 8 1 8 1 9 2 8 9 2 8 9 4 7 9 1 9 0 1 0 2

Буквы:

s d j e r d u u w d j k U M E D M O M J D I M O K S K D I D M E j d i s j d e I D I K Q S k e k d a s L l p f d o e o d o i k d w Q E E K K

Остальные символы:

\* # \* # \* @ # @ < ) ) # @ ( @ # \* ( ; ' ; ] ( # ( ) #

**Заключение:**

В процессе выполнения лабораторной работы было закреплено и освоено умение записывать данные из файла, считывать количество цифр, букв и символов из файла, сохраняя исходный порядок.