**МИНОБРНАУКИ РОССИИ**

**Санкт-Петербургский государственный**

**электротехнический университет**

**«ЛЭТИ» им. В.И. Ульянова (Ленина)**

**Кафедра МО ЭВМ**

отчет

**по лабораторной работе №3**

**по дисциплине «Алгоритмы и структуры данных».**

Тема: Стек. Очередь. Дек

Вариант 4 в.

|  |  |  |
| --- | --- | --- |
| Студент(ка) гр. 5383 |  | Допира В. Е. |
| Преподаватель |  | Фирсов М. А. |

Санкт-Петербург

2016

**Лабораторная работа 3**

**Стек. Очередь. Дек**

**Вариант 4**

**1. Цели**

1. Применить практические навыки работы со стеком и операций над ним;
2. Реализовать стек на базе вектора.

**Задача:** Содержимое заданного текстового файла F, разделенного на строки, переписать в текстовый файл G, выписывая литеры каждой строки в обратном порядке.

**2.Спецификация программы**

**2.1 Входные данные**

**Структуры данных:** в программе данные заданы с помощью структуры:

struct Str {

vector<string> lit ();

}

**2.2Выходные данные**

Программа выводит на экран и переписывает в файл литеры каждой строки в обратном порядке.

**3.Алгоритм и структуры данных**

Программа выполняет следующую последовательность действий:

|  |
| --- |
| ()  {  Чтение кирилицы  Выбор ввода данных: 0 - консоль, 1 - файл  Инициализация вектора и его размера  Ввод из консоли  {  Вывод сообщения: «Введите количество строк »  Чтение размеров вектора  Инициализация вектора, учитывая размер  Для (счетчика < длины вектора) { считывание литералов и помещение их в стек с помощью функции push\_back}  Реализация задания с помощью функции Rewrite()  }  Иначе{  Ввод из файла  Данные содержатся в "in.txt"  Если (файл не открыт)  {Вывод сообщения: «Файл input не может быть открыт!»  Выход из программы}  Считывание количества строк и вывод на экран  Инициализация вектора, учитывая размер  Для (счетчика < длины вектора) { Чтение из файла литералов и помещение их в стек с помощью функции push\_back  Вывод их на экран}  Реализация задания с помощью функции Rewrite()  }  Иначе Вывод сообщения: «Неверно выбран тип ввода данных!»  }  Вернуть 0  } |
| Функция переписывания литералов в обратном порядке(вектор)  {  Создаём файл для записи  Вывод сообщения: «В обратном порядке: »  Пока (счетчик >=0)  {  Для(Уменьшаем счетчик от (длины-1) до 0)  {Вывод на экран литералов и сообщений о добавлении их в файл  Запись строки литералов в файл  }  Удаление строки из стека с помощью функции pop\_back  Уменьшение счетчика  }  Вывод количества оставшихся элементов в стеке (для проверки)  Закрываем файл  } |

**Функции, используемые в программе**

void Rewrite(vector<string> lit)

Назначение: переписывание литеров каждой строки в обратном порядке

Параметры: vector<string> lit – вектор входных данных

Возвращаемое значение: нет.

**4. Набор тестов**

|  |  |  |  |
| --- | --- | --- | --- |
| **Входные данные** | | **Результат** | |
| **Длина** | **Строки** | **Ожидаемый** | **Полученный** |
| 1 | 12  45 | 54  21 | 54  21 |
| 2 | 15  abc  09 | 90  cba  51 | 90  cba  51 |
| 3 | 65723659125  зтсовшащомт  лс  4752470 | 0742574  сл  тмощашвостз  52195632756 | 0742574  сл  тмощашвостз  52195632756 |
| 4 | 83985948934  bvsdvjbsjdv  fvnsjvnjv  7608976p4587375823  op  тмыоамтфомтф | фтмофтмаоымт  po  3285737854p6798067  vjnvjsnvf  vdjsbjvdsvb  43984958938 | фтмофтмаоымт  po  3285737854p6798067  vjnvjsnvf  vdjsbjvdsvb  43984958938 |

1. **Текст программы:**

#include <iostream>

#include <fstream>

#include <string.h>

#include <vector>

using namespace std;

struct Str;

void Rewrite(vector<string> lit); //функция, реализующая задание

//структуры данных

struct Str {

vector<string> lit ();

};

int main()

{

setlocale(LC\_ALL,"Russian");

SetConsoleCP(866); //корректный вывод кириллицы

cout<<"Выберите тип ввода данных \n";

cout<<"0 - консоль, 1 - файл \n";

int enter=0;

cin>>enter;

int vecSize = 0; //размер вектора

vector<string> lit();

//ввод из консоли

if(enter == 0) {

cout<<"Введите количество строк ";

SetConsoleCP(1251);

cin >> vecSize;//размер вектора

SetConsoleCP(866);

vector<string> lit;

string str;

SetConsoleCP(1251);

for(int i = 0; i < vecSize; i++) {

cin >> str;

lit.push\_back(str);

}

Rewrite(lit);

}

else{

if(enter == 1) { //ввод из файла

ifstream fin("in.txt");

if (!fin.is\_open())

{

cerr << "Файл input не может быть открыт!" << endl;

return 1;

}

fin>>vecSize;

vector<string> lit;

string str;

cout<<"Количество строк: "<<vecSize<<endl;

for(int i = 0; i < vecSize; i++) { fin>>str;

lit.push\_back(str);

cout << str << ' \n ';}

Rewrite(lit);

}

else cerr<<"Неверно выбран тип ввода данных!\n";

}

return 0;

}

void Rewrite(vector<string> lit)

{

ofstream fout("out.txt"); // создаём объект класса ofstream для записи и связываем его с файлом

SetConsoleCP(866);

int i=lit.size()-1;

cout<<"\n В обратном порядке: "<<" \n";

while (i>=0)

{

for (int j=lit[i].size()-1; j>=0; j--)

{cout<<" добавлен элемент " <<lit[i][j] <<"\n";

fout<<lit[i][j]; // запись строки в файл

}

lit.pop\_back();

cout<<"Строка удалена из стека"<<" \n";

fout<<endl;

i--;

}

cout<<"\n элементов в стеке:"<<lit.size()<<" \n";

fout.close(); // закрываем файл

}

**6. Вывод**

В ходе данной работы были получены практические навыки работы с стеком на базе вектора на языке программирования С++.