МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное бюджетное образовательное учреждение высшего профессионального образования «Самарский государственный аэрокосмический университет имени академика С.П. Королёва (национальный исследовательский университет)» (СГАУ)

Факультет информатики

Кафедра программных систем

Дисциплина

**Объектно-ориентированное программирование**

**ОТЧЕТ**

по лабораторной работе № 2

<**Объектно-ориентированное программирование последовательного доступа к текстовым файлам с использованием функций стандартной библиотеки С++ в стиле языка C >**

Вариант № 9

Студент: Гижевская В.Д.

Группа: 6213-020302D

Преподаватель: Баландин А.В.

Оценка: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Дата: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Самара 2019

**Задание**

Написать программу, которая для заданного текстового файла, имя которого вводится с клавиатуры, строит в оперативной памяти массив записей (структур) вида, анализируя прочитанные строки 18 файла. Отсортировать полученный массив по возрастанию длин строк, результирующий массив вывести на экран. Для сортировки элементов массива использовать стандартную функцию в стиле языка С - qsort().

**Объектная модель**

Информационный объект – текстовый файл

* Свойства:
  + Поток доступа к файлу
  + Имя файла
  + Список записей
  + Список длин строк
* Процедуры:
  + Создание объекта файла
  + Формирование списка длин строк
  + Формирование списка записи
  + Сортировка списка записей
  + Вывод списка записей
  + Удаление объекта файла

Информационный объект – строка

* Свойства:
  + Длина строки
  + Количество строк такой длины

**Исходный текст программы**

**TextFile.h**

#define \_CRT\_SECURE\_NO\_WARNINGS

#pragma once

#include <iostream>

#include <string.h>

#include <stdio.h>

#include <stdlib.h>

#include <Windows.h>

typedef struct

{

int stringLength;

int stringNumber;

} String; //структура таблицы вывода

static int cmpf(const void\* \_a, const void\* \_b)//функция сравнения 2 записей

{

const String\* a = (String\*)\_a;

const String\* b = (String\*)\_b;

return a->stringLength - b->stringLength;

}

class TextFile

{

private:

FILE \*fp; //поток к файлу

int amount\_of\_strings; //количество строк

int\* mas\_of\_lengths; //массив длин строк

void make\_mas(); //создание массива записей

void make\_length\_mas();//создание массива длин строк

public:

char\* filename; //имя файла

String\* mas; //массив записей

friend int cmpf(const void\* first, const void\* second); //функция сравнения для сортировки

void sort\_mas(); //сортировка массива записей

void print\_table(); //вывод массива записей на консоль

TextFile(); // конструктор (ввод имени файла, создание массива)

~TextFile(); // деструктор

};

**TextFile.cpp**

#pragma once

#define \_CRT\_SECURE\_NO\_WARNINGS

#include "pch.h"

#include"TextFile.h"

//конструктор - создание объекта файла

TextFile::TextFile()

{

puts("Введите имя файла (не больше 20 символов):");

filename = (char\*)malloc(sizeof(char) \* 15);

amount\_of\_strings = 0; // число строк

scanf\_s("%s", filename);//считываем имя файла c консоли

strcat(filename, ".txt");

fp = fopen(filename, "rt");//открытие файла

if (fp == NULL)

{

puts("Ошибка открытия файла");

exit(1);

}

else

{

free(filename);

make\_length\_mas();

make\_mas();

}

}

//получение массива длин строк

void TextFile::make\_length\_mas()

{

char\* str = (char\*)malloc(100 \* sizeof(char));

mas\_of\_lengths = (int\*)malloc(sizeof(int) \* 50); // массив длин строк

while (fgets(str,99,fp) != NULL)

{

mas\_of\_lengths[amount\_of\_strings] = strlen(str);

amount\_of\_strings ++;

}

free(str);

fclose(fp);

}

//получение массива записей вида <длина строки, количество строк>

void TextFile::make\_mas()

{

mas = (String\*)malloc(sizeof(String) \* amount\_of\_strings); //массив записей

int same\_size;

for (int i = 0; i < amount\_of\_strings; i++)

{

same\_size = 0;

for (int ind = 0; ind < amount\_of\_strings; ind++)//подсчитывае количество строк каждой длины

{

if (mas\_of\_lengths[i] == mas\_of\_lengths[ind])

same\_size++;

}

mas[i] = { mas\_of\_lengths[i],same\_size };

}

free(mas\_of\_lengths);

}

//вывод на консоль массива записей

void TextFile::print\_table()

{

for (int m = 0; m < amount\_of\_strings; m++)

{

bool check = false;

for (int num = m - 1; num >= 0; num--)

{

if (mas[m].stringLength == mas[num].stringLength)

{

check = true;

break;

}

}

if(!check) printf("\t<%d,%d>\n", mas[m].stringLength, mas[m].stringNumber);

}

}

//сортировка массива записей

void TextFile:: sort\_mas()

{

qsort(mas, amount\_of\_strings,sizeof(String),cmpf);

}

//деструктор - удаление объекта файла

TextFile::~TextFile()

{

free(fp);

free(mas);

}

**OopTextFiles.cpp**

#define \_CRT\_SECURE\_NO\_WARNINGS

#pragma once

#include "pch.h"

#include <iostream>

#include<Windows.h>

#include"TextFile.h"

int main()

{

SetConsoleCP(1251);

SetConsoleOutputCP(1251);

setlocale(0, "");

puts("Лабораторная работа № 2\nГижевская Валерия\nВариант 9\n");

TextFile lr;

lr.sort\_mas();

puts("Массив записей вида <длина строк,количество строк>");

lr.print\_table();

system("pause\n");

return 0;

}