**Задание 1**

Реализовать меню, разработанное с применением массива указателей на функции.

#include <Windows.h>

#include <iostream>

using namespace std;

void division(float)

{

cout << "Введите два числа: \n";

int a, b;

cin >> a >> b;

cout<< "Результат равен: " << a / b;

}

void pow(float)

{

cout << "Введите два числа: \n";

int a, b;

cin >> a >> b;

cout<< "Результат равен: " << pow(a, b);

}

void sum(float)

{

cout << "Введите два числа: \n";

int a, b;

cin >> a >> b;

cout <<"Результат равен: "<< a + b;

}

int main()

{

setlocale(LC\_ALL, "ru");

SetConsoleCP(1251);

SetConsoleOutputCP(1251);

void (\*P[3])(float) = { division,pow,sum };

cout << "1.Деление двух чисел\n";

cout << "2.Возведение в степень\n";

cout << "3.Сумма двух чисел\n";

cout << " Какую из трех функций выбрать?\n ";

int my\_choose;

cin >> my\_choose;

my\_choose--;

(\*P[my\_choose])(my\_choose);

return 0;

}

**Задание 2**

Предварительно подготовить(программно или с помощью текстового редактора) текстовый файл,содержимым которого является массив (матрица) из чисел. В соответствии с индивидуальным заданием написать программу обработки содержимого файла.

Для каждого из вариантов задания необходимо выполнить следующие действия. Просмотреть содержимое исходного файла. Считать содержимое файла в одномерный (двумерный) динамический массив. Обработать

динамический массив согласно варианту задания. Полученный результат записать в конец исходного файла.

**Пример выполнения задания.** Отсортировать одномерный

целочисленный массив, находящийся в текстовом файле, методом «пузырька».Отсортированный массив дописать в конец исходного файла.

#include<iostream>

#include<string>

#include<fstream>

#include<Windows.h>

#include<vector>

using namespace std;

int main()

{

SetConsoleCP(1251);

SetConsoleOutputCP(1251);

setlocale(0, "ru");

string path = "myFile.txt";

ofstream file;

file.open(path); //не удаление предыдущих записей ofstream::app

const int n = 7;

int array[n];

if (!file.is\_open()) cout << "Error\n";

else

{

cout << "Данные записанные в файл: ";

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

array[i] = rand() % 10;

file << array[i] << " ";

cout << array[i] << " ";

}

cout << endl;

}

file.close();

ifstream fin;

fin.open(path);

if (!fin.is\_open()) cout << "Error\n";

else

{

char ch;

vector<char> myVector;

cout << "Данные взятые из файла: ";

while (fin.get(ch))

if (ch != ' ')

myVector.push\_back(ch);

for (int i = 0; i < myVector.size(); i++)

cout << myVector[i] << " ";

cout << endl;

cout << "Данные после сортировки пузырьком: ";

for (int i = 0; i < myVector.size(); i++) {

for (int j = 0; j < myVector.size() - 1; j++) {

if (myVector[j] > myVector[j + 1]) {

int t = myVector[j];

myVector[j] = myVector[j + 1];

myVector[j + 1] = t;

}

}

}

for (int i = 0; i < myVector.size(); i++)

cout << myVector[i] << " ";

file.open(path, ofstream::app);

file << endl;

if (!file.is\_open()) cout << "Error\n";

for (int i = 0; i < myVector.size(); i++) {

file << myVector[i] << " ";

}

file.close();

}

fin.close();

return 0;

}

3 Дано целое число K и текстовый файл. Вставить пустую строку перед строкой файла с номером K. Если строки с таким номером нет, то оставить файл без изменений.

#include<iostream>

#include<string>

#include<fstream>

#include<vector>

using namespace std;

int main()

{

setlocale(0, "ru");

ifstream fin;

fin.open("F.txt");

string str;

vector<string>myVector;

int number = 0;

while (!fin.eof())

{

str = "";

getline(fin, str);

myVector.push\_back(str);

cout << str << endl;

number++;

}

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

fin.close();

int s;

cout<<"После какой строки ставить пробел: ";

cin >> s;

if (s >= number)

cout << "Недостаточное кол-во строк" << endl;

cout << endl;

if (s < number)

{

for (int i = 0; i < myVector.size(); i++) {

if (i == s)

myVector.emplace(myVector.begin() + s, "\n");

}

}

ofstream fout;

fout.open("F.txt");

for (int i = 0; i < myVector.size(); i++)

if (i == s)

fout << myVector[i];

else

fout<< myVector[i] << endl;

fout.close();

return 0;

}

4 Дан символ C — прописная (заглавная) русская буква и текстовый файл. Создать строковый файл и записать в него все слова из исходного файла, начинающиеся на эту букву (прописную или строчную). Словом считать набор символов, не содержащий пробелов, знаков препинания и ограниченный пробелами, знаками препинания или началом/концом строки. Если исходный файл не содержит подходящих слов, то оставить результирующий файл пустым.

#include<iostream>

#include<string>

#include<fstream>

#include<Windows.h>

#include<vector>

using namespace std;

int main()

{

SetConsoleCP(1251);

SetConsoleOutputCP(1251);

setlocale(0, "ru");

ifstream fin;

fin.open("Laba.txt"); //не удаление предыдущих записей ofstream::app

if (!fin.is\_open()) cout << "Error\n";

else

{

cout << "Данные взятые из файла: " << endl;

string str;

while (!fin.eof())

{

str = "";

getline(fin, str);

cout << str << endl;

}

vector<string> myVector;

string temp;

for (int i = 0; i <str.length(); i++) {

if (str[i] == ' ') {

myVector.push\_back(temp);

temp.clear();

}

else if (i + 1 == str.length()) {

temp += str[i];

myVector.push\_back(temp);

}

else

temp += str[i];

}

ofstream fout;

fout.open("newLaba.txt");

for (int i = 0; i < myVector.size(); i++)

{

string s = myVector[i];

if (s[0] == 'с' || s[0] == 'С') {

fout << myVector[i]<<endl;

}

}

fout.close();

}

fin.close();

return 0;

}

5 Дан текстовый файл, каждая строка которого содержит изображения нескольких чисел, разделенные пробелами (вещественные числа имеют ненулевую дробную часть). Создать файл целых чисел, содержащий все целые числа из исходного файла (в том же порядке).

#include<iostream>

#include<string>

#include<fstream>

#include<Windows.h>

#include<vector>

using namespace std;

int main()

{

SetConsoleCP(1251);

SetConsoleOutputCP(1251);

setlocale(0, "ru");

ifstream fin;

fin.open("newLab.txt");

if (!fin.is\_open()) cout << "Error\n";

else

{

vector<double> myVector;

while (!fin.eof()) {

double v;

fin >> v;

if(floor(v) == v)

myVector.push\_back(v);

}

ofstream fout;

fout.open("newLaba.txt");

for (int i = 0; i < myVector.size(); i++) {

fout << myVector[i]<<" ";

}

fout.close();

}

fin.close();

return 0;

}