Министерство образования и науки РФ

Севастопольский государственный университет

Институт информационных технологий и управления в технических сетях

Лабораторная работа №6

**Потоки. Исключения.**

Выполнил:

ст.гр.ИСб-22д

Воронин И.Ю.

Проверил:

Тимофеев И.С.

Севастополь

2015

1.Цель работы

Изучить механизм потокового ввода/вывода, обеспечивающий гибкий и эффективный с гарантией типа метод обработки символьного ввода целых чисел, чисел с плавающей точкой и символьных строк, а также простую модель его расширения для обработки типов, определяемых пользователем. Приобрести практические навыки написания объектно-ориентированных программ с использованием обработки исключительных ситуаций.

2.Вариант задания

***Вариант 10***

1. Написать программу, печатающую символы от “*A*” до введенного с клавиатуры символа (последний возможный “*Z*”). Для каждого символа вывести номер, сам символ, шестнадцатеричный, восьмеричный и двоичный код этого символа. Сохранить результат работы программы в файле. Предусмотреть обработку ошибок.
2. Создать класс МойФайл, содержащий строку – путь к файлу. Описать перехват ошибок, связанных с некорректной работой с файлом. Пример: попытка открыть файл, не существующий по данному пути.

3.Текст программы

#pragma once

#include <iostream>

using namespace std;

class binary

{

unsigned long k;

public:

binary(unsigned long k) : k(k) {}

friend ostream& operator<<(ostream& os, const binary& t);

};

inline ostream& operator<<(ostream& os, const binary& t)

{

const unsigned long MAX = numeric\_limits<unsigned long>::max();

unsigned long bit = ~(MAX >> 1); // старший бит

while (bit) { os << (t.k & bit ? '1' : '0'); bit >>= 1; }

return os;

}

#pragma once

#include <string>

#include <fstream>

#include <iostream>

#include <io.h>

class MyFile

{

std::string path;

std::ofstream fout;

std::ifstream fin;

public:

MyFile();

MyFile(const char\*);

void Write();

void Read();

~MyFile();

};

#include "MyFile.h"

int file\_exist(const char\* name){

return \_access(name,0);

}

MyFile::MyFile(){

path = "";

std::cout << std::endl << "'File.txt' was open for writing." << std::endl;

}

MyFile::MyFile(const char\* \_path){

path = \_path;

try{

if (\_access(\_path, 0) == -1) { throw "Not\_found"; }

else{

}

}

catch(char\* ex) {

char st[20] = "";

std::cout << "1";

std::cout << "Error- file not found" << std::endl;

do{

std::cout << "Enter another path: ";

gets\_s(st);

} while (file\_exist(st) == -1);

path = st;

}

system("PAUSE");

}

void MyFile::Write(){

if (path != ""){

fout.open(path);

}

else{

fout.open("File.txt");

}

std::string str;

std::cout << "Input in file..." << std::endl << std::endl;

while (!(std::cin >> str));

fout << str;

fout.close();

system("PAUSE");

}

void MyFile::Read(){

setlocale(0, "RUS");

if (path!= ""){

fin.open(path);

}

else{

fin.open("File.txt");

}

if (fin.is\_open()){

std::cout << "NULL";

}

std::cout << std::endl << "Fie output..." << std::endl;

char str[256]="";

while (!fin.eof()){

fin.getline(str,256);

puts(str);

}

fin.close();

system("PAUSE");

}

MyFile::~MyFile(){

if (fin.is\_open()) fin.close();

if (fout.is\_open()) fout.close();

}

#pragma once

#include <iostream>

#include <fstream>

#include <iomanip>

#include "Binary.h"

#include "Letter.h"

#include "MyFile.h"

using namespace std;

extern void letter();

int main(){

//letter();

//MyFile mf("D:/1.txt");

MyFile mf("File.11txt");

mf.Read();

system("PAUSE");

return 0;

}

void letter(){

char ch;

ofstream fout;

fout.open("file.txt");

do{

system("CLS");

std::cout << "Enter letter('A'....'Z'): ";

try{

if (!(cin >> ch)) throw "Exeption(^Z)";

if (cin.rdbuf()->in\_avail() > 1) throw "Exception(to many symblos) ";

}

catch (char\* ex){

cout << ex << " was found." << endl;

cin.ignore(cin.rdbuf()->in\_avail());

cin.clear();

ch = 'a';

system("PAUSE");

}

} while (!(ch >= 'A' && ch <= 'Z'));

for (int i = 0; i <(int)ch - (int)'A' + 1; i++){

std::cout << setw(2) << dec << (i + 1) << " ";

fout << setw(2) << dec << (i + 1) << " ";

std::cout << (char)('A' + i) << " ";

fout << (char)('A' + i) << " ";

std::cout << std::oct << (int)('A' + i) << " ";

fout << std::oct << (int)('A' + i) << " ";

std::cout << std::hex << (int)('A' + i) << " ";

fout << std::hex << (int)('A' + i) << " ";

std::cout << binary((int)('A' + i)) << " ";

fout << binary((int)('A' + i)) << " ";

std::cout << std::endl;

fout << endl;

}

fout.close();

system("PAUSE");

}

4.Выполнения программы

Была написана программа, которая выполняла две различных задачи.

Первая состояла в том, чтобы при вводе буквы в заданном диапазоне печатать таблицу кодов, которая также бы записывалась в файл. К тому же данные действия должны были бы обрабатываться при помощи исключений.

Вторая задача состояла в написании класса, который работает с ввода-выводом данных в файл. Особенность класса в том, что в конструкторе при помощи обработки исключений проверялось, есть ли такой файл на диске.

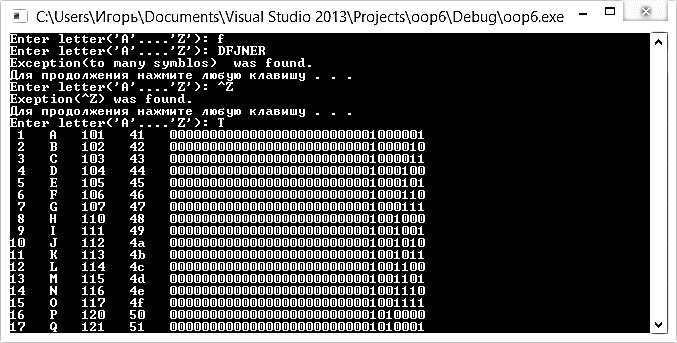


Рисунок 4.1. -Тест первой части программы.

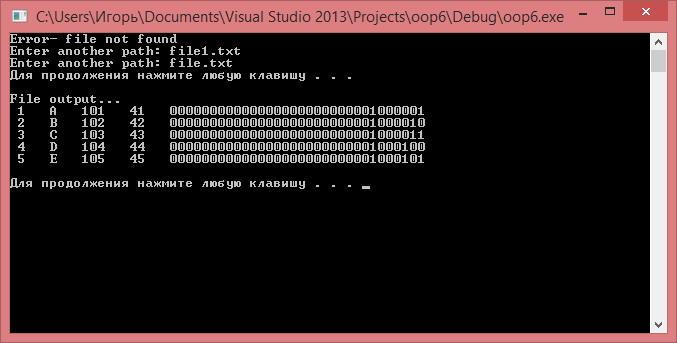


Рисунок 4.2. -Тест второй части программы.

ВЫВОДЫ

В данной лабораторной работе были изучены понятия потоков ввода вывода на экран, а также работы с файлами. Также данные программы писались с учётом обработки исключительных ситуаций. Действия каждой из подпрограмм сохранялись в текстовый файл, которые программа также могла считать. Программа была написана с использованием объектно-ориентированного программирования на языке С++.