Задача:

Выполнить расчет для двух написанных ранее программ (вы разрешили мне одну из-за большого кол-ва строк), но предусмотреть ввод исходных данных с клавиатуры.

Код:

.h

#pragma once

#ifndef UnifiedProblem1

#define UnifiedProblem1

void main();

void SwtchFunction(int Index);

void CalculationOf13DegreeNumber();

void ReturnLastDigitOfTheNumber();

void SumOf2To20();

void FactN();

void TheSwap();

void Sum1To100();

#endif

.cpp

#include <iostream>

//Для русского языка

#include <fstream>

#include “UnifiedProblem1.h”

#include <Windows.h>

using namespace std;

void main()

{

//Для русского языка

SetConsoleCP(1251);

SetConsoleOutputCP(1251);

int Index = -1;

cout << «Добро пожаловать!» << endl;

while (Index != 0 )

{

cout << endl;

cout << «Выберите программу из списка:» << endl;

cout << «1) Дано число x. Вычислить число x13 при помощи пяти операций умножения. Число из файла» << endl;

cout << «2) Дано натуральное число. Вывести его последнюю цифру. Число из файла» << endl;

cout << «3) Вычислить сумму четных чисел от 2 до n. N вводится из файла.» << endl;

cout << «4) Вычислить p = n! При n введенной из файла.» << endl;

cout << «5) Поменять местами значения переменных a и b. A и b из файла.» << endl;

cout << «6) Вычислить s = 1 + 2 + 3 + … + n. N из файла.» << endl;

cout << «Или нажмите 0 для завершения программы.» << endl << endl;

cin >> Index;

if (Index == 0) break;

SwtchFunction(Index);

Index = -1;

}

}

void SwtchFunction(int Index)

{

switch (Index)

{

case 1:

CalculationOf13DegreeNumber();

break;

case 2:

ReturnLastDigitOfTheNumber();

break;

case 3:

SumOf2To20();

break;

case 4:

FactN();

break;

case 5:

TheSwap();

break;

case 6:

Sum1To100();

break;

default:

cout << endl;

cout << «Введено неверное значение! Повторите попытку!» << endl;

break;

}

}

// Циклы 1

// Дано число x. Вычислите число x13 при помощи пяти операций умножения.

Void CalculationOf13DegreeNumber()

{

cout << endl << “Ввожу число x…” << endl;

int X, XforLoop, Xinput;

ifstream file(“A:\\UniversityStuff\\3Semester\\TechProg\\Problem2\\UnifiedProblem1\\Debug\\1.txt”);

if (!file || file.eof())

{

cout << endl << “Файл (1.txt) не найден или поврежден!” << endl;

return;

}

file >> Xinput;

cout << endl << “Число x равно “ << Xinput<< endl;

X = Xinput;

cout << endl << “Вычисляю 13 степень числа x…” << endl;

for (int Index = 0; Index <= 2; ++Index)

{

XforLoop = X;

X \*= XforLoop;

}

X \*= XforLoop;

X \*= Xinput;

cout << endl << «Ответ: x в 13 степени равен « << X << endl;;

}

//Циклы 2

//Дано натуральное число. Выведите его последнюю цифру.

Void ReturnLastDigitOfTheNumber()

{

cout << endl << «Ввожу натуральное число из файла…» << endl;

int Input;

ifstream file(“A:\\UniversityStuff\\3Semester\\TechProg\\Problem2\\UnifiedProblem1\\Debug\\1.txt”);

if (!file || file.eof())

{

cout << endl << “Файл (1.txt) не найден или поврежден!” << endl;

return;

}

for(int Index = 0; Index < 2; Index++) file >> Input;

cout << endl << «Натуральное число из файла равно « << Input << endl;

cout << endl << “Вычисляю…” << endl;

Input = Input % 10;

cout << endl << “Ответ: последня цифра – “<< Input << endl;

}

// Последовательности 2

//Вычислить сумму четных чисел от 2 до 20.

Void SumOf2To20()

{

int Sum = 0, Border1, Border2;

cout << endl << «Ввожу границы из файла…» << endl;

ifstream file(“A:\\UniversityStuff\\3Semester\\TechProg\\Problem2\\UnifiedProblem1\\Debug\\1.txt”);

if (!file || file.eof())

{

cout << endl << “Файл (1.txt) не найден или поврежден!” << endl;

return;

}

for (int Index = 0; Index < 3; Index++) file >> Border1;

file >> Border2;

if (Border2 < Border1) swap(Border1, Border2);

cout << endl << «Вычисляю сумму четных чисел из границ от « << Border1 << « до « << Border2 << endl;

for (int Index = Border1 + (Border1 % 2); Index <= 20; Index += 2)

{

Sum += Index;

}

cout << endl<< “Ответ: сумма равна “ << Sum << endl;

}

// Последовательности 1.7

// Вычислить p = n! При n = 8.

Void FactN()

{

int Fact = 1, n;

cout << endl << “Ввожу n из файла…” << endl;

ifstream file(“A:\\UniversityStuff\\3Semester\\TechProg\\Problem2\\UnifiedProblem1\\Debug\\1.txt”);

if (!file || file.eof())

{

cout << endl << “Файл (1.txt) не найден или поврежден!” << endl;

return;

}

for (int Index = 0; Index < 5; Index++) file >> n;

cout << endl << “n из файла равен “ << n << endl;

cout << endl << “Вычисляю факториал…” << endl;

for (int Index = 1; Index <= n; Index++)

{

Fact \*= Index;

}

cout << endl << “Ответ: факториал равен “ << Fact << endl;

}

// Циклы 4

// Даны две целые переменные a и b. Составить программу, после работы которой значения переменных поменялись

//бы местами, но не используя каких-либо других дополнительных переменных

void TheSwap()

{

int A, B;

cout << endl << “Ввожу a и b из файла…” << endl;

ifstream file(“A:\\UniversityStuff\\3Semester\\TechProg\\Problem2\\UnifiedProblem1\\Debug\\1.txt”);

if (!file || file.eof())

{

cout << endl << “Файл (1.txt) не найден или поврежден!” << endl;

return;

}

for (int Index = 0; Index < 6; Index++) file >> A;

file >> B;

cout << endl << “Исходно a = “ << A << “, b = “ << B << “\n”;

cout << endl << «Меняю местами…» << endl;

swap(A, B);

cout << endl << “Ответ: теперь a = “ << A << “, b = “ << B << endl;

}

//Последовательности 1. Вычислить s = 1 + 2 + 3 + … +100.

Void Sum1To100()

{

//TODO интерфейс для юзера

int Sum = 0, Border;

cout << endl << «Ввожу правую границу для суммы из файла…» << endl;

ifstream file(“A:\\UniversityStuff\\3Semester\\TechProg\\Problem2\\UnifiedProblem1\\Debug\\1.txt”);

if (!file || file.eof())

{

cout << endl << “Файл (1.txt) не найден или поврежден!” << endl;

return;

}

for (int Index = 0; Index < 8; Index++) file >> Border;

cout << endl << «Правая граница для суммы из файла равна « << Border << endl;

cout << endl << “Суммирую…” << endl;

for (int Index = 1; Index <= Border; Index++)

{

Sum += Index;

}

cout << endl << “Ответ: сумма равна “ << Sum << endl;}

Расчет:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Имя характеристики | Количество с учетом сложности | | | |
| Низкий | Средний | Высокий | Итого |
| Внешние вводы | 1\*3= 3 | 0\*4= 0 | 0\*6= 0 | 3 |
| Внешние выводы | 0\*4= 0 | 50\*5= 250 | 0\*7= 0 | 250 |
| Внешние запросы | 6\*3= 0 | 0\*4= 0 | 0\*6= 0 | 24 |
| Внутренние логические файлы | 29\*7= 203 | 0\*10= 0 | 0\*15 = 0 | 203 |
| Внешние интерфейсные файлы | 1\*5= 5 | 0\*7= 0 | 0\*10= 0 | 5 |
| Общее количество | | | | 485 |

|  |  |  |
| --- | --- | --- |
| N | Вопрос для системного параметра | Обозна-чение |
| 1 | Какое влияние имеет наличие средств передачи данных? | f1 = 5 |
| 2 | Какое влияние имеет распределенная обработка данных? | f2 = 1 |
| 3 | Какое влияние имеет распространенность используемой аппаратной платформы? | f3 = 1 |
| 4 | Какое влияние имеет критичность к требованиям производительности и ограничению времени ответа? | f4 = 1 |
| 5 | Какое влияние имеет частота транзакций? | f5 = 1 |
| 6 | Какое влияние имеет ввод данных в режиме реального времени? | f6 = 3 |
| 7 | Какое влияние имеет эффективность работы конечного пользователя? | f7 = 5 |
| 8 | Какое влияние имеет оперативное обновление локальных файлов в режиме реального времени? | f8 = 1 |
| 9 | Какое влияние имеет скорость обработки данных (вычислений)? | f9 = 1 |
| 10 | Какое влияние имеет количество и категории пользователей? | f10 = 1 |
| 11 | Какое влияние имеет легкость инсталляции? | f11 = 1 |
| 12 | Какое влияние имеет легкость эксплуатации? | f12 = 1 |
| 13 | Какое влияние имеет разнообразие условия применения? | f13 = 1 |
| 14 | Какое влияние имеет простота внесения изменений? | f14 = 1 |

Сумма fi = 24

Расчёт Function Points: FP= общее количество ×(0,65+0,01×fi) = 485×(0,65+0,01×24) =**431,65**.