Міністерство освіти і науки України

Центральноукраїнський національний технічний університет

Механіко-технологічний факультет

ЗВІТ

ПРО ВИКОНАННЯ ЛАБОРАТОРНОЇ РОБОТИ № 10

з навчальної дисципліни

“Базові методології та технології програмування”

Реалізація програмних модулів оброблення даних складових типів з файловим введенням/виведенням

ЗАВДАННЯ ВИДАВ

доцент кафедри кібербезпеки та програмного забезпечення

Доренський О. П.

[https://github.com/odorenskyi/](https://github.com/odorenskyi/Dmytro-Parkhomenko-KB18)

ВИКОНАВ

студент академічної групи КІ22-2

Ткаченко О. Ю.

ПЕРЕВІРИВ

ст. викладач кафедри кібербезпеки   
та програмного забезпечення

Собінов О. Г.

Кропивницький – 2023

**Мета:** Набуття ґрунтовних вмінь і практичних навичок реалізації у Code::Blocks IDE мовою програмування С++ програмних модулів створення й оброблення даних типів масив, структура, об’єднання, множина, перелік, перетворення типів даних, використання файлових потоків та функцій стандартних бібліотек для оброблення символьної інформації.

**Варіант №10(Файл L10-Var-019.jpg)**

**Завдання**: 1. Реалізувати програмні модулі розв’язування задач 10.1–10.3 як складові статичної бібліотеки libModulesПрізвище.а (проект ModulesПрізвище лабораторних робіт №8–9).

2. Реалізувати тестовий драйвер автоматизованої перевірки програмних модулів розв’язування задач 10.1–10.3.

Вихідний код програм можна знайти в Додатках Б, В

Юніт тести можна знайти в Додатку А

Результати роботи тест драйвера:

Вхідний файл не існує

Test suite #1: passed

Вхідний файл не існує

Test suite #2: passed

Test suite #3: passed

Test driver status: passed

**Висновок:** Виконавши цю лабораторну роботу з теми “ Реалізація програмних модулів оброблення даних складових типів з файловим введенням/виведенням” я набув ґрунтовних вмінь і практичних навичок реалізації у Code::Blocks IDE мовою програмування С++ програмних модулів створення й оброблення даних типів масив, структура, об’єднання, множина, перелік, перетворення типів даних, використання файлових потоків та функцій стандартних бібліотек для оброблення символьної інформації.

Як і в попередній лабораторній роботі, я отримав практичний досвід використання статичних бібліотек в проектах та розуміння важливості їх застосування в розробці програмного забезпечення, так як наприклад першу функцію s\_calculation я зміг використати повторно без написання нового коду. Також я отримав досвід розширення та підтримки вже створеного модуля.

Я набув цінних знань і вмінь, які дозволять мені ефективно працювати з файлами у своїх програмах. Також, я ознайомився з концепцією файлових потоків, які є зручним способом зчитування і запису даних у файли. Це дозволяє мені взаємодіяти з зовнішніми файлами, зберігати і отримувати дані для подальшої обробки. навчився відкривати, закривати і переміщуватись по файлам за допомогою різних методів і функцій, що надає мова C++. Це дозволяє мені контролювати доступ до файлової системи і здійснювати потокову обробку даних.

Ця лабораторна робота набагато складніша, ніж попередні, було складно писать як самі модулі, так і тест драйвер. Відчувається деякий стрибок в складності задач.

**ДОДАТОК А**

|  |  |  |
| --- | --- | --- |
| Preliminary Steps | Action (test steps) | Expected Result |
| Ім’я вхідного файлу iFile.cntu (створюється і заповнюється автоматично тест драйвером)  Ім’я вихідного файлу oFile.cntu  (створюється модулем) | task10\_1\_fFunc(“iFile.cntu”, “oFile.cntu”) | Вміст вихідного файлу:  Розробник: Олексiй Ткаченко  Органiзацiя: ЦНТУ  Мiсто, Країна: Кропивницький, Україна  Рiк розробки: 2023  (випадкове число від 10 до 100) - випадкове число від 10 до 100  Вхідний файл не містить пунктуаційних помилок |
| Ім’я вхідного файлу not\_a\_file  (не існує)  Ім’я вхідного файлу oFile.cntu  (створюється модулем) | task10\_1\_fFunc(“not\_a\_file”, “oFile.cntu”) | Тест драйвер виведе в консоль:  Вхідний файл не існує |

Unit tests 10.1

|  |  |  |
| --- | --- | --- |
| Preliminary Steps | Action (test steps) | Expected Result |
| Ім’я вхідного файлу iFile.cntu (створюється і заповнюється автоматично тест драйвером) | task10\_2\_fFunc(“iFile.cntu”) | Вміст вхідного файлу:  Як парость виноградної лози, плекайте мову.  Пильно й ненастанно політь бур'ян.  Чистіша від сльози вона хай буде.  Вірно і слухняно нехай вона щоразу служить вам,  Хоч і живе своїм живим життям.  (Дата запуску тестового драйвера) |
| Ім’я вхідного файлу not\_a\_file  (не існує) | task10\_2\_fFunc(“not\_a\_file”) | Тест драйвер виведе в консоль:  Вхідний файл не існує |

Unit tests 10.2

|  |  |  |
| --- | --- | --- |
| Preliminary Steps | Action (test steps) | Expected Result |
| Ім’я вихідного файлу oFile.cntu  (створюється модулем) | task10\_3\_fFunc(“oFile.cntu”, 1.2, -0.2, 2.0, 580299) | Вміст вихідного файлу:  Розробник: Олексiй Ткаченко  Органiзацiя: ЦНТУ  Мiсто, Країна: Кропивницький, Україна  Рiк розробки: 2023  (випадкове число від 10 до 100) - випадкове число від 10 до 100  Вхідний файл не містить пунктуаційних помилок  11.04 - результат роботи функції s\_calculation()  10001101101011001011 |
| Ім’я вихідного файлу oFile.cntu  (створюється модулем) | task10\_3\_fFunc(“oFile.cntu”, -1.2, -2.0, 3.3, 12546474412) | Вміст вихідного файлу:  Розробник: Олексiй Ткаченко  Органiзацiя: ЦНТУ  Мiсто, Країна: Кропивницький, Україна  Рiк розробки: 2023  (випадкове число від 10 до 100) - випадкове число від 10 до 100  Вхідний файл не містить пунктуаційних помилок  11.04 - результат роботи функції s\_calculation()  1000110110101100101111  4.8 - результат роботи функції s\_calculation()  1011101011110101000000000110101100 |

Unit tests 10.3

**ДОДАТОК Б**

#include <iostream>

#include <fstream>

#include <ModulesTkachenkoOY.h>

#include <ctime>

using namespace std;

int main()

{

system("chcp 1251");

system("cls");

string wrongName = "not\_a\_file";

string oFileName = "oFile.cntu";

string iFileName = "iFile.cntu";

bool passed = true;

char randomCheck=0;

string line = "";

// input file creation

try{

ofstream inputFileCreate(iFileName, std::ios\_base::out | std::ios::binary);

if(!inputFileCreate.is\_open()){throw(0);}

inputFileCreate << "Як парость виноградної лози, плекайте мову.\n"

"Пильно й ненастанно політь бур'ян.\nЧистіша від сльози вона хай буде.\n"

"Вірно і слухняно нехай вона щоразу служить вам,\n"

"Хоч і живе своїм живим життям.\n";

inputFileCreate.close();

}

catch(int errCode){

cout<< "Завершення роботи: Не вдалось створити необхідний файл";

getchar();

return 0;

}

// first module;

try{

for(int i=0;i<3;i++){

if(!task10\_1\_fFunc(iFileName, oFileName)){throw(0);}

ifstream task10\_1(oFileName, std::ios\_base::in | std::ios::binary);

for(int counter = 0; counter<5; counter++){

getline(task10\_1, line);

if(counter==4){

if(stoi(line) >=10 && stoi(line) <=100){

randomCheck++;

}

}

}

task10\_1.close();

}

if(randomCheck<2){passed = false;}

if(!task10\_1\_fFunc(wrongName, oFileName)){throw(1);}

passed = false;

}

catch(int errCode){

if(!errCode){

passed = false;

} else if(errCode && randomCheck>=2) {

cout<<"Test suite #1: passed\n";

}

}

//second module

try{

char s[12];

time\_t t = time(0);

strftime(s, 12, "%d.%m.%Y", localtime(&t));

if(!task10\_2\_fFunc(iFileName)){throw(0);}

ifstream task10\_2(iFileName, std::ios\_base::in | std::ios::binary);

for(int counter = 0; counter<6; counter++){

getline(task10\_2, line);

if(counter==5){

if(!(line == s)){

passed = false;

}

}

}

task10\_2.close();

if(!task10\_2\_fFunc(wrongName)){throw(1);}

passed = false;

}

catch(int errCode){

if(!errCode){

passed = false;

} else if (errCode && passed){

cout<<"Test suite #2: passed\n";

}

}

//third module

try{

if(!task10\_3\_fFunc(oFileName, 1.2, -0.2, 2.0, 580299)){throw(0);}

ifstream task10\_3(oFileName, std::ios\_base::in | std::ios::binary);

for(int counter = 0; counter<8; counter++){

getline(task10\_3, line);

if(counter==6){

if(!(stof(line) == s\_calculation(1.2, -0.2, 2.0))){

passed = false;

}

} else if(counter == 7){

if (!(line == "10001101101011001011")){

passed = false;

}

}

}

task10\_3.close();

if(!task10\_3\_fFunc(oFileName, -1.2, -2.0, 3.3, 12546474412)){throw(0);}

task10\_3.open(oFileName, std::ios\_base::in | std::ios::binary);

for(int counter = 0; counter<10; counter++){

getline(task10\_3, line);

if(counter==8){

if(!(stof(line) == s\_calculation(-1.2, -2.0, 3.3))){

passed = false;

}

} else if(counter == 9){

if (!(line == "1011101011110101000000000110101100")){

passed = false;

}

}

}

task10\_3.close();

cout<< "Test suite #3: passed\n";

}

catch(int errCode){

passed = false;

}

cout << "Test driver status: ";

if(passed){

cout << "passed\n";

} else{ cout<< "failed\n"; }

cout<< "Для завершення натисніть Enter\n";

getchar();

return 0;

}

**ДОДАТОК В**

#include <cmath>

#include <iostream>

#include <iomanip>

#include <fstream>

#include <string>

#include <ctime>

#include <bitset>

#include <cstdlib>

#define PI 3.14

struct results{

float totalWage, processedWage, totalTax;

std::string sockSizeEU;

int sockSizeUS, numberOfBits, typeOfBits;

};

float s\_calculation(float x, float y, float z){

float s = round(pow( 2\*z+1 , x) \* 100)/100 - round(sqrt(abs(y - z/2))\*100)/100 + z + PI;

return s;

}

results wagePaid(int a = 0, int b = 0, int c = 0, int d = 0 , int e = 0)

{

int workHours[5] {a, b, c, d, e};

int totalWage;

float totalTax;

if(a == 0 && b == 0 && c == 0 && d == 0 && e == 0)

std::cout << "Послiдовно введiть робочi години з понедiлка по п'ятницю, роздiляючи їх натисканнями Enter" <<std::endl;

for(int i=0;i<5;i++){

if(workHours[i] == 0){

std::cin >> workHours[i];

}

if(workHours[i] > 8){

totalWage+=300;

}

totalWage+=workHours[i]\*40;

}

totalTax = totalWage \* 0.186;

results wageResults;

wageResults.totalWage = totalWage;

wageResults.totalTax = totalTax;

wageResults.processedWage = totalWage - totalTax;

return wageResults;

};

results maleSocksSizeConvert(int sockSize = 0){

results sockSizes;

if (sockSize == 0){

std::cout << "Введiть розмiр шкарпеток(примiтка: цiле число вiд 23 до 31)" << std::endl;

std::cin >> sockSize;

}

if(sockSize>=23 && sockSize<=31){

switch (sockSize){

case 23:

sockSizes.sockSizeEU = "37/38"; sockSizes.sockSizeUS = 8;

break;

case 25:

sockSizes.sockSizeEU = "39/40"; sockSizes.sockSizeUS = 9;

break;

case 27:

sockSizes.sockSizeEU = "41/42"; sockSizes.sockSizeUS = 10;

break;

case 29:

sockSizes.sockSizeEU = "43/44"; sockSizes.sockSizeUS = 11;

break;

case 31:

sockSizes.sockSizeEU = "45/46"; sockSizes.sockSizeUS = 12;

break;

}

}

return sockSizes;

}

results bitInNumber(int num = -1){

results bitResults;

int counter = 0;

int N7;

if(num == -1){

std::cout << "Введiть цiле число вiд 0 до 7483650"<<std::endl;

std::cin >> num;

}

N7 = (num >> 6)& 1;

if(N7){

while (num) {

counter += !(num & 1);

num >>= 1;

}

bitResults.typeOfBits = 0;

} else{

while (num) {

counter += num & 1;

num >>= 1;

}

bitResults.typeOfBits = 1;

}

bitResults.numberOfBits = counter;

return bitResults;

}

int task10\_1\_fFunc(std::string iFileName, std::string oFileName){

srand((int)time(0));

try{

std::ifstream iFile(iFileName, std::ios\_base::in | std::ios::binary);

if(!iFile.is\_open()){throw(1);}

std::ofstream oFile(oFileName, std::ios\_base::out | std::ios::binary);

if(!oFile.is\_open()){throw(0);}

oFile << "Розробник: Олексiй Ткаченко\nОрганiзацiя: ЦНТУ\nМiсто, Країна: Кропивницький, Україна\nРiк розробки: 2023\n"

<< (rand()%90)+10 << " - випадкове число від 10 до 100"<<'\n';

std::string s;

std::string myText;

std::string sArr[300];

std::string pcArr[6] {",", ".", ".", ".", ",", "."};

while (getline (iFile, myText)) {

s = s + myText;

}

int arrCounter = 0;

std::string buffer;

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

if(s[i] == ','){

sArr[arrCounter] = buffer;

arrCounter++;

buffer.clear();

sArr[arrCounter] = ",";

arrCounter++;

} else if(s[i] == '.'){

sArr[arrCounter] = buffer;

arrCounter++;

buffer.clear();

sArr[arrCounter] = ".";

arrCounter++;

} else if(s[i] == ' '){

continue;

} else{

buffer +=s[i];

}

}

bool isErr = false;

for(int pcArrCounter = 0, arrCounter = 1; pcArrCounter<6;pcArrCounter++, arrCounter+=2){

if(sArr[arrCounter] != pcArr[pcArrCounter]){

isErr = true;

break;

}

}

if(isErr){

oFile << "Вхідний файл містить пунктуаційну помилку"<<'\n';

}else {oFile << "Вхідний файл не містить пунктуаційних помилок" << '\n';}

oFile.close();

iFile.close();

return 1;

}

catch(int erNum){

switch(erNum){

case 0:

std::cout << "Не вдалось створити файл\n";

break;

case 1:

std::cout << "Вхідний файл не існує\n";

break;

}

return 0;

}

}

int task10\_2\_fFunc(std::string iFileName){

char s[12];

time\_t t = time(0);

strftime(s, 12, "%d.%m.%Y", localtime(&t));

try{

std::ifstream itFile(iFileName, std::ios\_base::in);

if(!itFile.is\_open()){throw(0);}

itFile.close();

std::ofstream iFile(iFileName, std::ios\_base::app | std::ios::binary);

if(!iFile.is\_open()){throw(0);}

iFile << s << '\n';

iFile.close();

return 1;

}

catch(int erNum){

std::cout << "Вхідний файл не існує\n";

return 0;

}

}

std::string binary {""};

void binFunc(size\_t b)

{

if (b > 1)

binFunc(b / 2);

binary += std::to\_string(b % 2);

}

int task10\_3\_fFunc(std::string oFileName, float x = 0, float y = 0, float z = 0, size\_t b = 0){

if(x == 0){

std::cout << "Введіть x(дійсне число): " << '\n';

std::cin >> x;

}

if(y == 0){

std::cout << "Введіть y(дійсне число): " << '\n';

std::cin >> y;

}

if(z == 0){

std::cout << "Введіть z(дійсне число): " << '\n';

std::cin >> z;

}

if(b == 0){

std::cout << "Введіть b(натуральне число): " << '\n';

std::cin >> b;

}

try{

std::ofstream oFile(oFileName, std::ios\_base::app | std::ios::binary);

if(!oFile.is\_open()){throw(0);}

binFunc(b);

oFile << s\_calculation(x,y,z)<<" - результат роботи функції s\_calculation()" << '\n' << binary << '\n';

oFile.close();

binary = "";

return 1;

}

catch(int erNum){

std::cout << "Не вдалось створити або відкрити файл\n";

return 0;

}

}