Міністерство освіти і науки України

Центральноукраїнський національний технічний університет

Механіко-технологічний факультет

Кафедра кібербезпеки та програмного забезпечення

# 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

ВИКОНАВ

студент академічної групи

КІ-21-1 Алєксєєвський Владислав

ПЕРЕВІРИВ

викладач кафедри кібербезпеки

та програмного забезпечення

П.С. Усік

Кропивницький – 2022

# Лабораторна робота №10

**Тема:** Реалізація програмних модулів оброблення даних складових типів з файловим введенням/виведенням.

**Мета роботи:** полягає у набутті 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 15

**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.

# ХІД РОБОТИ

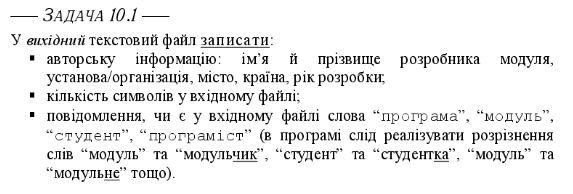


Рисунок 1 - Умова задачі 10.1

### 

### **Строга постановка задачі:**

Вхідні дані: текст в форматі string записане до вхідного файлу.

Вихідні дані: записані до вихідного файлу: авторська інформація, кількість символів та результат пошуку слів в тексті.

### **Проектування програмного модуля:**

**authorInfo()** - приймає в якості аргументу назву вихідного файлу. Виконується очищення вмісту файлу та запис авторської інформації.  
**symbolsCountInFile ()** - отримує на вхід назву вихідного файлу та текст, що міститься у вхідному файлі. Цикл ітерації якого дорівнює довжині тексту, реалізовано для перевірки наявності символу переносу рядка. На вихід отримуємо дозапис у вихідний файл кількості символів в тексті.

**findWordInText ()** - отримує на вхід назву вихідного файлу та текст, що міститься у вхідному файлі. Текст ініціалізований як рядок, котрий надалі використовується в функції бібліотеки <сstring> для пошуку вхідного слова. Істинність пошуку визначається співпадінням позиції каретки у файлі с позицією результату стандартної функції find(). До вихідного файлу записується результат пошуку.

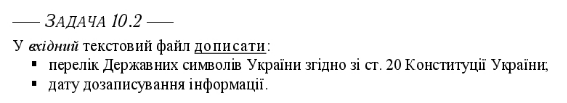


Рисунок 2 - Умова задачі 10.2

### **Строга постановка задачі:**

Вхідні дані: текст в форматі string записане до вхідного файлу .

Вихідні дані: записані до вихідного файлу: перелік Державних символів України, дата та час дозапису.

**Проектування програмного модуля:**

**numPunctuationMarksInFile ()** - приймає в якості аргументу назву вихідного файлу. У вхідний файл ведеться дозапис кількості знаків пунктуації в ньому.

**timestampInFile()** - приймає аргумент назву вхідного файлу. В даний файл записується дата та час дозапису інформації за допомогою стандартної функції з бібліотеки <ctime>.

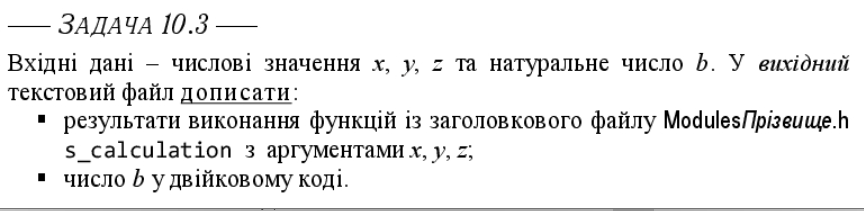


Рисунок 3 - Умова задачі 10.3

### **Строга постановка задачі:**

Вхідні та вихідні дані вичерпно описані в умові завдання.

### **Проектування програмного модуля:**

**sCalculationResInFile()** - аргументи: назва вихідного файлу та три дійсних числа. До вихідного файлу записується результат виконання функції s\_caculation, що приймає в якості аргументів три дійсні значення.

**decimalToBinary()** - вхідні дані: назва вихідного файлу та натуральне число. Оголошується масив розміром в 32 елементи (стільки бітів вміщує змінна типу int) і в циклі, з використанням оператору ‘%’ заповнюємо массив остачами від ділення на 2. В наступному циклі виводимо поелементно, починаючи з останнього, вміст масиву до вихідного файлу.

### **Результат виконання TestDriver.exe:**

Testcase 1

Запис слова до вхідного файлу: true

10.1(1) - Запис авторської інформації до вихідного файлу: true

10.1(2) - Результат підрахунку кількості голосних літер та дозапис до вихідного файлу: true

10.1(3) - Результат пошуку слова та дозапис до вихідного файлу: true

10.2(1) - Результат підрахунку кількості приголосних літер та дозапис до вхідного файлу: true

10.2(2) - Доапис часу редагування файлу до вхідного файлу: true

10.3(1) - Дозапис результату функції s\_calculation до вихідного файлу: true

10.3(2) - Результат конвертування десяткового числа в двійкове та дозапис до вихідного файлу: true

=======================================================

Press any key to enter next test key

=======================================================

Testcase 2

Запис слова до вхідного файлу: true

10.1(1) - Запис авторської інформації до вихідного файлу: true

10.1(2) - Результат підрахунку кількості голосних літер та дозапис до вихідного файлу: true

10.1(3) - Результат пошуку слова та дозапис до вихідного файлу: true

10.2(1) - Результат підрахунку кількості приголосних літер та дозапис до вхідного файлу: true

10.2(2) - Доапис часу редагування файлу до вхідного файлу: true

10.3(1) - Дозапис результату функції s\_calculation до вихідного файлу: true

10.3(2) - Результат конвертування десяткового числа в двійкове та дозапис до вихідного файлу: true

=======================================================

Press any key to enter next test key

=======================================================

Testcase 3

Запис слова до вхідного файлу: true

10.1(1) - Запис авторської інформації до вихідного файлу: true

10.1(2) - Результат підрахунку кількості голосних літер та дозапис до вихідного файлу: true

10.1(3) - Результат пошуку слова та дозапис до вихідного файлу: true

10.2(1) - Результат підрахунку кількості приголосних літер та дозапис до вхідного файлу: true

10.2(2) - Доапис часу редагування файлу до вхідного файлу: true

10.3(1) - Дозапис результату функції s\_calculation до вихідного файлу: true

10.3(2) - Результат конвертування десяткового числа в двійкове та дозапис до вихідного файлу: true

=======================================================

Press any key to enter next test key

=======================================================

Testcase 4

Запис слова до вхідного файлу: true

10.1(1) - Запис авторської інформації до вихідного файлу: true

10.1(2) - Результат підрахунку кількості голосних літер та дозапис до вихідного файлу: true

10.1(3) - Результат пошуку слова та дозапис до вихідного файлу: true

10.2(1) - Результат підрахунку кількості приголосних літер та дозапис до вхідного файлу: true

10.2(2) - Доапис часу редагування файлу до вхідного файлу: true

10.3(1) - Дозапис результату функції s\_calculation до вихідного файлу: true

10.3(2) - Результат конвертування десяткового числа в двійкове та дозапис до вихідного файлу: true

=======================================================

Press any key to enter next test key

=======================================================

Testcase 5

Запис слова до вхідного файлу: true

10.1(1) - Запис авторської інформації до вихідного файлу: true

10.1(2) - Результат підрахунку кількості голосних літер та дозапис до вихідного файлу: true

10.1(3) - Результат пошуку слова та дозапис до вихідного файлу: true

10.2(1) - Результат підрахунку кількості приголосних літер та дозапис до вхідного файлу: true

10.2(2) - Доапис часу редагування файлу до вхідного файлу: true

10.3(1) - Дозапис результату функції s\_calculation до вихідного файлу: true

10.3(2) - Результат конвертування десяткового числа в двійкове та дозапис до вихідного файлу: true

=======================================================

End of Testcases

=======================================================

**Висновок:**

Проаналізувавши завдання лабораторної роботи та користуючись поняттям модульності програмного забезпечення було прийнято рішення про градуювання кожного завдання на підфункції, що сконцентровані на вирішенні конкретної, логічної частини завдання. Метою розбиття функцій є досягнення більш детального декларування результатів тестування окремих частин програми.

До вихідного коду статичної бібліотеки, створеної під час виконання лабораторної роботи №8 та №9, було додано реалізації функцій для розв’язування задач 10.1-10.3. Після компіляції проекту бібліотеки в теці \obj оновився файл з розширенням .а, що і є файлом статичної бібліотеки. Наступним кроком було додавання до заголовкового файлу бібліотеки прототипів функцій.

Для виведення результатів виконання функцій зі статичної бібліотеки labModulesAlieksieievskyi.a у консольне вікно застосунку TestDriver.exe було використано маніпулятор потокового виводу boolalpha із стандартної бібліотеки <iomanip>. Таким чином, підтверджується рівність очікуваних даних з отриманими у випадку, якщо функція повертає значення не типу bool. функції, з результатом повернення bool визначають, чи є, насамперед, істинним виконання даної функції, чи ні.

Враховано коректне відкриття та закриття файлових потоків для всіх функцій. Так, на початку функціонування кожної процедури оголошується змінна типу, відповідного до мети взамодії файлових потоків (ofstream - для запису / ifstream - для читання). Додатково, в аргументах даних змінних, окрім вказання назви файлу, вказано тип відкриття файлу (ios::out - для запису / ios::in - для зчитування / ios::app - для дозапису). В режимі ios::out працюють лише ті процедури, що мають виконати перезапис всього файлу (тобто функції, що вперше взаємодіють з файлом).

Аби запобігти повторенню однакової частини коду в кожній функції, на початку функціонування TestDriver.exe викликається fileEditing(), що перевіряє обидва файли на можливість до відкриття / редагування.

Також, для автоматизації процесу була опущена умова лабораторної роботи щодо необхідності запису будь-якого українського тексту до вхідного файлу власноруч. Натомість реалізовано додаткову процедуру, що за прийнятим аргументом в якості будь-якого українського тексту виконує його запис до вхідного файлу.

Пошук символів у тексті здійснюється завдяки перевірці кожного елементу рядка чи дорівнює він одному зі символів. Надалі виконується порівняння з константними значеннями результату. Якщо умова виконується - пошук вважається успішним.

Функція визначення двійкового представлення числа основана на стандартному (“ручному”) методі переведення числа з десяткової системи числення до двійкової. Врахувавши, що число типу int вміщує в собі 32 біти, оголошуємо масив з відповідним розміром для його подальшого заповнення двійковими 0 та 1. В першому циклі вхідне число поступово ділиться на 2 з отриманням остачі (використовується оператор %) та ділиться на 2 звичайним методом для подальшої ітерації. Цикл завершується, коли числа стає рівне 0. Так як біти в числі рахуються зправа-наліво, створено наступний цикл, в котрому відбувається поелементний запис до вихідного файлу елементів масиву, починаючи з останнього і закінчуючи початковим.

Результат unit test–у всіх тест кейсів є Passed. Автоматизоване тестування проведено успішно.

**ДОДАТОК А**

(TestSuite до завдання 10.1)

|  |  |  |
| --- | --- | --- |
|  | Назва тестового набору /  **Test Suite Description** | TS\_10.1 |
|  | Назва проекта/ПЗ /  **Name of project** | TestDriver.ехе |
|  | Рівень тестування /  **Level of testing** | Модульний |
|  | Автор тест-сьюта /  **Test Suite Author** | Алєксєєвський Владислав |
|  | Виконавець /  **Implementer** | Алєксєєвський Владислав |

|  |  |  |  |
| --- | --- | --- | --- |
| **Test Case ID** | **Actions** | **Expected Result** | **Test Result** |
| TS\_1 | 1. Запис  "Коли в 1968 році я став працювати викладачем факультету журналістики Київського університету, Матвія Шестопала там уже не було, його викинули з роботи, заплямували ім’я різними — ізмами, залишили без шматка хліба і можливості його заробляти." у prjInputFile.txt  2. Запис інформації у prjOutputFile.txt  3. Запис кількості символів у prjOutputFile.txt  4. Запис пошуку слів “студент” “модуль” “програміст” “програма”у prjOutputFile.txt | **Вміст prjInputFile.txt:**  Коли в 1968 році я став працювати викладачем факультету журналістики Київського університету, Матвія Шестопала там уже не було, його викинули з роботи, заплямували ім’я різними — ізмами, залишили без шматка хліба і можливості його заробляти.  **Вміст prjOutputFile.txt:**  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  \* Виконавець: Алєксєєвський Владислав Владиславович \*  \* ВНЗ: Центральний Національний Технічний Університет \*  \* Місто/Країна: Кропивницький/Україна \*  \* Рік розробки: 2022 \*  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  Кількість cимволів з вхідного файлу: 38  Cлово <<програма>> - не знайдено  Cлово <<програміст>> - не знайдено  Cлово <<модуль>> - не знайдено  Cлово <<студент>> - не знайдено | **Passed** |
| TS\_2 | 1. Запис  " Мав єдину зустріч із ним десь на початку сімдесятих років, зустріч на вулиці, біля університетської бібліотеки, під час якої інший викладач факультету журналістики (В.В.Полковенко) і познайомив мене." у prjInputFile.txt  2. Запис інформації у prjOutputFile.txt  3. Запис кількості символів у prjOutputFile.txt  4. Запис пошуку слів “студент” “модуль” “програміст” “програма”у prjOutputFile.txt | **Вміст prjInputFile.txt:**  Мав єдину зустріч із ним десь на початку сімдесятих років, зустріч на вулиці, біля університетської бібліотеки, під час якої інший викладач факультету журналістики (В.В.Полковенко) і познайомив мене**.**  **Вміст prjOutputFile.txt:**  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  \* Виконавець: Алєксєєвський Владислав Владиславович \*  \* ВНЗ: Центральний Національний Технічний Університет \*  \* Місто/Країна: Кропивницький/Україна \*  \* Рік розробки: 2022 \*  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  Кількість cимволів з вхідного файлу: 34  Cлово <<програма>> - не знайдено  Cлово <<програміст>> - не знайдено  Cлово <<модуль>> - не знайдено  Cлово <<студент>> - не знайдено | **Passed** |
| TS\_3 | 1. Запис  " Кілька хвилин розмови — і яскрава пам’ять про нього на все життя: енергійний, ерудований, комунікабельний." у prjInputFile.txt  2. Запис інформації у prjOutputFile.txt  3. Запис кількості символів у prjOutputFile.txt  4. Запис пошуку слів “студент” “модуль” “програміст” “програма”у prjOutputFile.txt | **Вміст prjInputFile.txt:**  Кілька хвилин розмови — і яскрава пам’ять про нього на все життя: енергійний, ерудований, комунікабельний**.**  **Вміст prjOutputFile.txt:**  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  \* Виконавець: Алєксєєвський Владислав Владиславович \*  \* ВНЗ: Центральний Національний Технічний Університет \*  \* Місто/Країна: Кропивницький/Україна \*  \* Рік розробки: 2022 \*  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  Кількість cимволів з вхідного файлу: 18  Cлово <<програма>> - не знайдено  Cлово <<програміст>> - не знайдено  Cлово <<модуль>> - не знайдено  Cлово <<студент>> - не знайдено | **Passed** |
| TS\_4 | 1. Запис  " Пізніше, коли натрапив у книзі мемуарів Іллі Еренбурга згадку про зустріч із Матвієм Шестопалом, офіцером, учасником війни 1941–1945 років, я глибше зрозумів цю людину, а враження доповнювали його колишні студенти, які говорили про нього лише із високим пієтетом, а 17 грудня 1997 р. влаштували прекрасний вечір пам’яті свого вчителя у Спілці письменників, хоча членом Спілки письменників України Матвій Шестопал не був." у prjInputFile.txt  2. Запис інформації у prjOutputFile.txt  3. Запис кількості символів у prjOutputFile.txt  4. Запис пошуку слів “студент” “модуль” “програміст” “програма”у prjOutputFile.txt | **Вміст prjInputFile.txt:**  Пізніше, коли натрапив у книзі мемуарів Іллі Еренбурга згадку про зустріч із Матвієм Шестопалом, офіцером, учасником війни 1941–1945 років, я глибше зрозумів цю людину, а враження доповнювали його колишні студенти, які говорили про нього лише із високим пієтетом, а 17 грудня 1997 р. влаштували прекрасний вечір пам’яті свого вчителя у Спілці письменників, хоча членом Спілки письменників України Матвій Шестопал не був.  **Вміст prjOutputFile.txt:**  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  \* Виконавець: Алєксєєвський Владислав Владиславович \*  \* ВНЗ: Центральний Національний Технічний Університет \*  \* Місто/Країна: Кропивницький/Україна \*  \* Рік розробки: 2022 \*  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  Кількість cимволів з вхідного файлу: 70  Cлово <<програма>> - не знайдено  Cлово <<програміст>> - не знайдено  Cлово <<модуль>> - не знайдено  Знайдено слово <<студент>> | **Passed** |
| TS\_5 | 1. Запис  " Це було пошанування недожитого вісімдесятиріччя видатної людини." у prjInputFile.txt  2. Запис інформації у prjOutputFile.txt  3. Запис кількості символів у prjOutputFile.txt  4. Запис пошуку слів “студент” “модуль” “програміст” “програма”у prjOutputFile.txt | **Вміст prjInputFile.txt:**  Це було пошанування недожитого вісімдесятиріччя видатної людини**.**  **Вміст prjOutputFile.txt:**  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  \* Виконавець: Алєксєєвський Владислав Владиславович \*  \* ВНЗ: Центральний Національний Технічний Університет \*  \* Місто/Країна: Кропивницький/Україна \*  \* Рік розробки: 2022 \*  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  Кількість cимволів з вхідного файлу: 7  Cлово <<програма>> - не знайдено  Cлово <<програміст>> - не знайдено  Cлово <<модуль>> - не знайдено  Cлово <<студент>> - не знайдено | **Passed** |

**ДОДАТОК Б**

(TestSuite до завдання 10.2)

|  |  |  |
| --- | --- | --- |
|  | Назва тестового набору /  **Test Suite Description** | TS\_10.2 |
|  | Назва проекта/ПЗ /  **Name of project** | TestDriver.ехе |
|  | Рівень тестування /  **Level of testing** | Модульний |
|  | Автор тест-сьюта /  **Test Suite Author** | Алєксєєвський Владислав |
|  | Виконавець /  **Implementer** | Алєксєєвський Владислав |

|  |  |  |  |
| --- | --- | --- | --- |
| **Test Case ID** | **Actions** | **Expected Result** | **Test Result** |
| TS\_1 | 1. Запис державних символів України InputFile.txt  2. Запис часу дозапису у InputFile.txt | **Вміст prjInputFile.txt:**  Стаття 20 Конституції України визначає, що державними символами України є Державний Прапор України, Державний Герб України і Державний Гімн України.  Дата та час дозапису інформаціїї: [date] | **Passed** |
| TS\_2 | 1. Запис державних символів України InputFile.txt  2. Запис часу дозапису у InputFile.txt | **Вміст prjInputFile.txt:**  Стаття 20 Конституції України визначає, що державними символами України є Державний Прапор України, Державний Герб України і Державний Гімн України.  Дата та час дозапису інформаціїї: [date] | **Passed** |
| TS\_3 | 1. Запис державних символів України InputFile.txt  2. Запис часу дозапису у InputFile.txt | **Вміст prjInputFile.txt:**  Стаття 20 Конституції України визначає, що державними символами України є Державний Прапор України, Державний Герб України і Державний Гімн України.  Дата та час дозапису інформаціїї: [date] | **Passed** |
| TS\_4 | 1. Запис державних символів України InputFile.txt  2. Запис часу дозапису у InputFile.txt | **Вміст prjInputFile.txt:**  Стаття 20 Конституції України визначає, що державними символами України є Державний Прапор України, Державний Герб України і Державний Гімн України.  Дата та час дозапису інформаціїї: [date] | **Passed** |
| TS\_5 | 1. Запис державних символів України InputFile.txt  2. Запис часу дозапису у InputFile.txt | **Вміст prjInputFile.txt:**  Стаття 20 Конституції України визначає, що державними символами України є Державний Прапор України, Державний Герб України і Державний Гімн України.  Дата та час дозапису інформаціїї: [date] | **Passed** |

**ДОДАТОК В**

(TestSuite до завдання 10.3)

|  |  |  |
| --- | --- | --- |
|  | Назва тестового набору /  **Test Suite Description** | TS\_10.3 |
|  | Назва проекта/ПЗ /  **Name of project** | TestDriver.ехе |
|  | Рівень тестування /  **Level of testing** | Модульний |
|  | Автор тест-сьюта /  **Test Suite Author** | Алєксєєвський Владислав |
|  | Виконавець /  **Implementer** | Алєксєєвський Владислав |

|  |  |  |  |
| --- | --- | --- | --- |
| **Test Case ID** | **Actions** | **Expected Result** | **Test Result** |
| TS\_1 | 1. Запис результату s\_calculation (2, 3, -1) у outputFile.txt  2. Запис двійкового числа 5 у outputFile.txt | **Вміст outputFile.txt:**  **[…]**  Результат виконання s\_calculation: -9.91247  Двійкове представлення числа 8 : 1000 | **Passed** |
| TS\_2 | 1. Запис результату s\_calculation (7, -5, 1) у outputFile.txt  2. Запис двійкового числа 5 у outputFile.txt | **Вміст outputFile.txt:**  **[…]**  Результат виконання s\_calculation: -206.233  Двійкове представлення числа 17 : 1001 | **Passed** |
| TS\_3 | 1. Запис результату s\_calculation (10, 20, -0) у outputFile.txt  2. Запис двійкового числа 5 у outputFile.txt | **Вміст outputFile.txt:**  **[…]**  Результат виконання s\_calculation: 0.00136931  Двійкове представлення числа 71 : 100111 | **Passed** |
| TS\_4 | 1. Запис результату s\_calculation (1, -2, 4) у outputFile.txt  2. Запис двійкового числа 5 у outputFile.txt | **Вміст outputFile.txt:**  **[…]**  Результат виконання s\_calculation: 1.16005  Двійкове представлення числа 139 : 10001011 | **Passed** |
| TS\_5 | 1. Запис результату s\_calculation (39, 9, -4) у outputFile.txt  2. Запис двійкового числа 5 у outputFile.txt | **Вміст outputFile.txt:**  **[…]**  Результат виконання s\_calculation: 10359.9  Двійкове представлення числа 612 : 1001100100 | **Passed** |

**ДОДАТОК Г**

(Вихідний код завдання 10.1, 10.2, 10.3)

**… / ModulesAlieksieievskyi/ main.cpp**

#include <math.h>

#include <iostream>

#include <sstream>

#include <bitset>

#include <ctime>

#include <fstream>

#include "ModulesAlieksieievskyi.h"

//Lab 8

//Task 8

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

{

return (sin(z)\*(pow(x,2)\*y)+sqrt(fabs(z - 12\*x))/ pow(y,3));

}

//Lab 9

//Task 9.1

string ratingExplanation (int rating)

{

if (rating >= 1 && rating <= 34)

{

return "Your grade : FX Explanation : With the obligatory repeated course";

}

else if (rating >= 35 && rating <= 59)

{

return "Your grade : F Explanation : With the possibility of re-assembly";

}

else if (rating >= 60 && rating <= 66)

{

return "Your grade : E Explanation : Satisfied minimum criteria";

}

else if (rating >= 67 && rating <= 74)

{

return "Your grade : D Explanation : Not bad, but with significant numbers of defects";

}

else if (rating >= 75 && rating <= 81)

{

return "Your grade : C Explanation : Overall correct fulfillment of certain significant numbers of errors";

}

else if (rating >= 82 && rating <= 89)

{

return "Your grade : B Explanation : Above average some mistakes";

}

else if (rating >= 90 && rating <= 100)

{

return "Your grade : A Explanation : Excellent performance fish small number errors";

}

}

//Task 9.2

cAndF temperatureCAndF(float sumWeekTempC)

{

float weaklyTempC = floor((sumWeekTempC/7)\*10)/10;

float weaklyTempF = floor(((sumWeekTempC/7)\*1.8 + 32)\*10)/10;

stringstream descripiton;

descripiton << "Average weekly temperature on the Celsius scale: "

<< weaklyTempC

<< ", Fahrenheit scale: "

<< weaklyTempF;

return {weaklyTempC, weaklyTempF, descripiton.str()};

}

//Task 9.3

int bitsNumber(int number)

{

bitset<32> b\_number{number};

if (b\_number[12])

{

return 32 - b\_number.count();

}

return b\_number.count();

}

//Lab\_10

void fileEditing(string outputFileName, string inputFileName)

{

ofstream inputFile(inputFileName);

ofstream outputFile(outputFileName);

if (!inputFile || !outputFile) {

cout << "Неможливо відкрити файл для редагування\a" << endl;

inputFile.close();

outputFile.close();

exit(1);

}

inputFile.close();

outputFile.close();

}

bool fillInputTxtFile(string inputFileName, string ukrText)

{

ofstream inputFile(inputFileName, ios::out);

inputFile << ukrText << endl << endl;

inputFile.close();

return true;

}

//Task 10.1(1)

bool authorInfo(string outputFileName)

{

ofstream outputFile(outputFileName, ios::out);

outputFile << "\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*" << endl

<< "\* Виконавець: Алєксєєвський Владислав Владиславович \*" << endl

<< "\* ВНЗ: Центральний Національний Технічний Університет \*" << endl

<< "\* Місто/Країна: Кропивницький/Україна \*" << endl

<< "\* Рік розробки: 2022 \*" << endl

<< "\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*" << endl << endl;

outputFile.close();

return true;

}

//Task 10.1(2)

int symbolsCountInFile(string outputFileName, string ukrText)

{

ofstream outputFile(outputFileName, ios::app);

char symbols[31] = { '-', '+', '=', ')', '(', '/', '\*', '{', '}', '[', ']', '<', '>', '\_', '`', '|', '^', '~', ';', ':', '$', '#', '@', '"', '.', ',', '!', '&', '?', ' '};

int symbolsCount = 0;

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

for(int j = 0; j < 31; j++)

if (symbols[j] == ukrText[i]) {

symbolsCount++;

}

}

outputFile << "Кількість cимволів з вхідного файлу: " << symbolsCount << endl << endl;

outputFile.close();

return symbolsCount;

}

//Task 10.1(3)

bool findWordInText(string outputFileName, string ukrText)

{

ofstream outputFile(outputFileName, ios::app);

size\_t foundWord1 = ukrText.find("програма");

size\_t foundWord2 = ukrText.find("програміст ");

size\_t foundWord3 = ukrText.find("модуль");

size\_t foundWord4 = ukrText.find("студент");

if (foundWord1 != string::npos)

outputFile << "Знайдено слово <<програма>>" << endl << endl;

else

outputFile << "Cлово <<програма>> - не знайдено" << endl << endl;

if (foundWord2 != string::npos)

outputFile << "Знайдено слово <<програміст>>" << endl << endl;

else

outputFile << "Cлово <<програміст>> - не знайдено" << endl << endl;

if (foundWord3 != string::npos)

outputFile << "Знайдено слово <<модуль>>" << endl << endl;

else

outputFile << "Cлово <<модуль>> - не знайдено" << endl << endl;

if (foundWord4 != string::npos)

outputFile << "Знайдено слово <<студент>>" << endl << endl;

else

outputFile << "Cлово <<студент>> - не знайдено" << endl << endl;

outputFile.close();

return true;

}

//Task 10.2(1)

bool stateSymbolInConstitution(string inputFileName)

{

ofstream inputTxtFile(inputFileName, ios::app);

inputTxtFile << "Стаття 20 Конституції України визначає, що державними символами України є Державний Прапор України, Державний Герб України і Державний Гімн України." << endl

<< "Державний Прапор України – стяг із двох рівновеликих горизонтальних смуг синього і жовтого кольорів." << endl << endl;

inputTxtFile.close();

return true;

}

//Task 10.2(2)

bool timestampInFile(string inputFileName)

{

ofstream inputFile(inputFileName, ios::app);

time\_t rawtime;

time(&rawtime);

inputFile << "Дата та час дозапису інформаціїї: " << ctime(&rawtime) << endl << endl;

inputFile.close();

return true;

}

//Task 10.3(1)

double sCalculationResInFile(string outputFileName, float x, float y, float z)

{

ofstream outputFile(outputFileName, ios::app);

outputFile << "Результат виконання s\_calculation: " << s\_calculation(x, y, z) << endl << endl;

outputFile.close();

return s\_calculation(x, y, z);

}

//Task 10.3(2)

bool decimalToBinary(string outputFileName, unsigned int number)

{

if (number <= 0) {

return false;

}

ofstream outputFile(outputFileName, ios::app);

int binaryCels[32];

int i;

unsigned int convertedNum = number;

for(i = 0; convertedNum > 0; i++) {

binaryCels[i] = convertedNum % 2;

convertedNum = convertedNum / 2;

}

outputFile << "Двійкове представлення числа " << number << " : ";

for(i = i - 1; i >= 0; i--) {

outputFile << binaryCels[i];

}

outputFile << endl << endl;

outputFile.close();

return true;

}

**… / ModulesAlieksieievskyi/ ModulesAlieksieievskyi.h**

#ifndef MODULESALIEKSIEIEVSKYI\_H\_INCLUDED

#define MODULESALIEKSIEIEVSKYI\_H\_INCLUDED

#include <math.h>

#include <string>

using namespace std;

struct cAndF

{

float weaklyTempC;

float weaklyTempF;

std::string description = "";

};

//Lab 8

float s\_calculation (int x, int y, int z);

//Lab 9

string ratingExplanation (int rating);

cAndF temperatureCAndF(float sumWeekTemp);

int bitsNumber(int number);

//Lab 10

void fileEditing(string outputFileName, string inputFileName);

bool fillInputTxtFile(string inputFileName, string ukrText);

bool authorInfo(string outputFileName);

int symbolsCountInFile(string outputFileName, string ukrText);

bool findWordInText(string outputFileName, string ukrText);

bool stateSymbolInConstitution(string inputFileName);

bool timestampInFile(string inputFileName);

double sCalculationResInFile(string outputFileName, float x, float y, float z);

bool decimalToBinary(string outputFileName, unsigned int number);

#endif // MODULESALIEKSIEIEVSKYI\_H\_INCLUDED;

**ДОДАТОК Д**

**… / TestDriver / main.cpp**

#include <iostream>

#include <cstring>

#include <conio.h>

#include <iomanip>

#include <windows.h>

#include "ModulesAlieksieievskyi.h"

using namespace std;

int main()

{

system ("chcp 1251 & cls");

const string inputWordsArray[5] = { "Коли в 1968 році я став працювати викладачем факультету журналістики Київського університету, Матвія Шестопала там уже не було, його викинули з роботи, заплямували ім’я різними — ізмами, залишили без шматка хліба і можливості його заробляти",

"Мав єдину зустріч із ним десь на початку сімдесятих років, зустріч на вулиці, біля університетської бібліотеки, під час якої інший викладач факультету журналістики (В.В.Полковенко) і познайомив мене.",

"Кілька хвилин розмови — і яскрава пам’ять про нього на все життя: енергійний, ерудований, комунікабельний.",

"Пізніше, коли натрапив у книзі мемуарів Іллі Еренбурга згадку про зустріч із Матвієм Шестопалом, офіцером, учасником війни 1941–1945 років, я глибше зрозумів цю людину, а враження доповнювали його колишні студенти, які говорили про нього лише із високим пієтетом, а 17 грудня 1997 р. влаштували прекрасний вечір пам’яті свого вчителя у Спілці письменників, хоча членом Спілки письменників України Матвій Шестопал не був.",

"Це було пошанування недожитого вісімдесятиріччя видатної людини." };

const int symbolsCount[5] = {38, 34, 18, 70, 7};

const float sCalcX[5] = { 2, 7, 10, 1, 39 };

const float sCalcY[5] = { 3, -5, 20, -2, 9 };

const float sCalcZ[5] = { -1, 1, 0, 4, -4 };

const unsigned int naturalDigit[5] = {8, 17, 71, 139, 612};

const double expectedResult[5] = {-9.91247, -206.233, 0.00136931, 1.16005, 10359.9};

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

fileEditing("OutputFile.txt", "InputFile.txt");

cout << "Testcase " << i + 1 << endl

<< boolalpha

<< "Запис слова до вхідного файлу: \t\t\t\t\t\t\t\t\t"

<< fillInputTxtFile("InputFile.txt", inputWordsArray[i]) << endl;

cout << boolalpha

<< "10.1(1) - Запис авторської інформації до вихідного файлу: \t\t\t\t\t"

<< authorInfo("OutputFile.txt")

<< endl;

cout << boolalpha

<< "10.1(2) - Результат підрахунку кількості символів та дозапис до вихідного файлу: \t\t"

<< (symbolsCountInFile("OutputFile.txt", inputWordsArray[i]) == symbolsCount[i])

<< endl;

cout << boolalpha

<< "10.1(3) - Результат пошуку слова та дозапис до вихідного файлу: \t\t\t\t"

<< findWordInText("OutputFile.txt", inputWordsArray[i])

<< endl;

cout << boolalpha

<< "10.2(1) - Дозапис до вхідного файлу державних символів України: \t\t\t\t"

<< stateSymbolInConstitution("InputFile.txt")

<< endl;

cout << boolalpha

<< "10.2(2) - Доапис часу редагування файлу до вхідного файлу: \t\t\t\t\t"

<< timestampInFile("InputFile.txt")

<< endl;

cout << boolalpha

<< "10.3(1) - Дозапис результату функції s\_calculation до вихідного файлу: \t\t\t\t"

<< (sCalculationResInFile("OutputFile.txt", sCalcX[i], sCalcY[i], sCalcZ[i]) <= expectedResult[i] + 0.0005)

<< endl;

cout << boolalpha

<< "10.3(2) - Результат конвертування десяткового числа в двійкове та дозапис до вихідного файлу: \t"

<< decimalToBinary("OutputFile.txt", naturalDigit[i])

<< endl << endl;

if (i < 4){

cout << "\t\t=======================================================" << endl

<< "\t\t\tPress any key to enter next test key" << endl

<< "\t\t=======================================================" << endl << endl;

}else{

cout << "\t\t=======================================================" << endl

<< "\t\t\t\tEnd of Testcases" << endl

<< "\t\t=======================================================" << endl << endl;

}

getch();

}

return 0;

}