|  |  |
| --- | --- |
| Назва тестового набору  Test Suite Description | TS\_7\_1 |
| Назва проекта / ПЗ  Name of Project / Software | Honcharenko-task\_7\_1 |
| Рівень тестування  Level of Testing | системний / System Testing |
| Автор тест-сьюта  Test Suite Author |  |
| Виконавець  Implementer | Honcharenko Vlad |

|  |  |  |  |
| --- | --- | --- | --- |
| Ід-р тест-кейса / Test Case ID | Дії (кроки) /  Action  (Test Steps) | Очікуваний  результат /  Expected Result | Результат тестування (пройшов/не вдалося/ заблокований) /  Test Result (passed/failed/ blocked) |
| 1 | Введення: Це просто тестовий приклад. | Слово 'програма' не знайдено | passed |
| 2 | Введення: Цей текст не містить слово програма. | Слово 'програма' знайдено | passed |
| 3 | Введення: У програмуванні важлива логіка. | Слово 'програма' не знайдено | passed |
| 4 | Введення: Привіт! В мене є своя програма. | Слово 'програма' не знайдено | passed |
| 5 | Введення: Сьогодні, я розробив нову програму. | Слово 'програма' не знайдено | passed |
| 6 | Введення: програма для розробки програмного забезпечення! | Слово 'програма' знайдено | passed |
| 7 | Введення: веб-додаток, а не веб-програма! | Слово 'програма' не знайдено | passed |
| 8 | Введення: програма - це набір інструкцій, який визначає послідовність дій для виконання конкретного завдання або розв'язання певної проблеми на комп'ютері. | Слово 'програма' знайдено | passed |
| 9 | Введення: У мого товариша є програма для вирішення будь-якої задачі. | Слово 'програма' знайдено | passed |
| 10 | Введення: Програма для редагування фотографій виявилася дуже потужною і зручною у використанні. | Слово 'програма' не знайдено | passed |

|  |  |
| --- | --- |
| Назва тестового набору  Test Suite Description | TS\_7\_2 |
| Назва проекта / ПЗ  Name of Project / Software | Honcharenko-task\_7\_2 |
| Рівень тестування  Level of Testing | системний / System Testing |
| Автор тест-сьюта  Test Suite Author |  |
| Виконавець  Implementer | Honcharenko Vlad |

|  |  |  |  |
| --- | --- | --- | --- |
| Ід-р тест-кейса / Test Case ID | Дії (кроки) /  Action  (Test Steps) | Очікуваний  результат /  Expected Result | Результат тестування (пройшов/не вдалося/ заблокований) /  Test Result (passed/failed/ blocked) |
| 1 | Введіть 14 натуральних чисел:  1 2 3 4 5 6 7 8 9 10 11 12 13 14 | Кількість чисел рівних 150: 0  Кількість чисел рівних 1000: 0  Кількість чисел рівних 10000: 0 | passed |
| 2 | Введіть 14 натуральних чисел:  150 150 150 150 150 150 150 150 150 150 150 150 150 150 | Кількість чисел рівних 150: 14  Кількість чисел рівних 1000: 0  Кількість чисел рівних 10000: 0 | passed |
| 3 | Введіть 14 натуральних чисел:  10000 10000 10000 10000 10000 10000 10000 10000 10000 10000 10000 10000 10000 10000 | Кількість чисел рівних 150: 0  Кількість чисел рівних 1000: 0  Кількість чисел рівних 10000: 14 | passed |
| 4 | Введіть 14 натуральних чисел:  1000 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000 1000 | Кількість чисел рівних 150: 0  Кількість чисел рівних 1000: 14  Кількість чисел рівних 10000: 0 | passed |
| 5 | Введіть 14 натуральних чисел:  1000 150 1000 150 1000 150 1000 150 1000 150 1000 150 1000 150 | Кількість чисел рівних 150: 7  Кількість чисел рівних 1000: 7  Кількість чисел рівних 10000: 0 | passed |
| 6 | Введіть 14 натуральних чисел:  150 1000 10000 150 1000 10000 150 1000 10000 150 1000 10000 150 1000 | Кількість чисел рівних 150: 5  Кількість чисел рівних 1000: 5  Кількість чисел рівних 10000: 4 | passed |
| 7 | Введіть 14 чисел:  -5 150 -1000 0 -150 1000 -2000 -150 500 10000 -10000 150 -1000 2000 | Кількість чисел = 150: 3  Кількість чисел = 1000: 3  Кількість чисел = 10000: 2 | passed |
| 8 | Введіть 14 чисел:  0 0 0 0 0 0 0 0 0 0 0 0 0 0 | Кількість чисел = 150: 0  Кількість чисел = 1000: 0  Кількість чисел = 10000: 0 | passed |
| 9 | Введіть 14 чисел:  10 20 30 40 50 60 70 80 90 100 500 1000 10000 1250 | Кількість чисел = 150: 0  Кількість чисел = 1000: 1  Кількість чисел = 10000: 1 | passed |
| 10 | Введіть 14 чисел:  256 150 10000 512 1000 1024 150 4096 10000 150 1000 8192 150 2048 | Кількість чисел = 150: 5  Кількість чисел = 1000: 3  Кількість чисел = 10000: 3 | passed |

Міністерство освіти і науки України

Центральноукраїнський національний технічний університет

Механіко-технологічний факультет

ЗВІТ

ПРО ВИКОНАННЯ ЛАБОРАТОРНОЇ РОБОТИ № 7

з навчальної дисципліни

“Базові методології та технології програмування”

ПРОГРАМНА РЕАЛІЗАЦІЯ ОБРОБЛЕННЯ МАСИВІВ

ДАНИХ ТА СИМВОЛЬНОЇ ІНФОРМАЦІЇ ЗА СТАНДАРТОМ

UNICODE

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 КН-23

Гончаренко В.В.

ПЕРЕВІРИВ

ст. викладач кафедри кібербезпеки   
та програмного забезпечення

Ганна Дрєєва.

Кропивницький – 2024

ЛАБОРАТОРНА РОБОТА № 7

**ПРОГРАМНА РЕАЛІЗАЦІЯ ОБРОБЛЕННЯ МАСИВІВ**

**ДАНИХ ТА СИМВОЛЬНОЇ ІНФОРМАЦІЇ ЗА СТАНДАРТОМ**

**UNICODE**

*Мета роботи* полягає у набутті ґрунтовних вмінь і практичних навичок розроблення

циклічних алгоритмів розв’язування обчислювальних задач та їх формального

представлення мовою програмування С (ISO/IEC 9899:2018) задля реалізації

програмних засобів у вільному кросплатформовому Code::Blocks IDE.

**ЗАВДАННЯ ДО ЛАБОРАТОРНОЇ РОБОТИ**

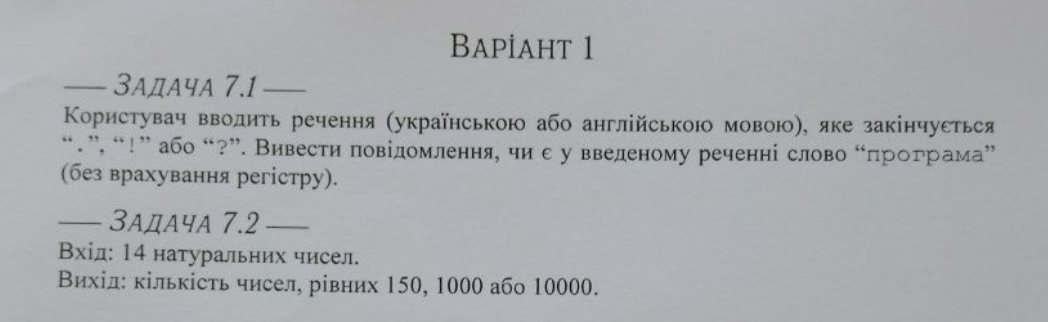
1. Створити персональний обліковий запис GitHub.

2. Реалізувати програмне забезпечення розв’язування задачі 7.1.

3. Реалізувати програмне забезпечення розв’язування задачі 7.2.

4. Створити Git-репозиторій для спільної роботи над проєктом з

контролем версій.



Кінець

Виведення

("Слово 'програма' не знайдено або

("Слово 'програма' не знайдено

Початок

Введення

MaxSentence, sentence, word, choice.

Початок циклу do-while.

[Ввід речення] (fgets).

[Виведення] "Введіть речення:" за допомогою printf

**- Задача 7.1 –**

#include <stdio.h>

#include <stdlib.h>

#include <string.h>

#include <ctype.h>

int main()

{

system("chcp 1251 & cls");

int MaxSentence = 100;

char sentence[MaxSentence];

char \*word = "програма";

char choice;

do {

printf("Введіть речення (для закінчення введіть: \",\" \".\" \"!\" \"?\"): ");

fflush(stdin);

fgets(sentence, MaxSentence, stdin);

sentence[strcspn(sentence, ",.?!")] = '\0';

for (int i = 0; sentence[i]; i++) {

sentence[i] = tolower(sentence[i]);

}

if (strstr(sentence, word) != NULL) {

printf("Слово 'програма' знайдено.\n");

} else {

printf("Слово 'програма' не знайдено.\n");

}

printf("Бажаєте ввести ще речення? (y/n): ");

scanf(" %c", &choice);

} while (choice == 'y' || choice == 'Y');

return 0;

}

**Результат:**

Введіть речення (для закінчення введіть: "," "." "!" "?"): Привіт! В мене є своя програма.

Слово 'програма' не знайдено.

**- Задача 7.2 –**

Кінець

Виведення

Кількість чисел = 150,  
Кількість чисел = 1000,

Кількість чисел = 10000

Початок

Введення

count\_150, count\_1000, count\_10000.

Початок циклу for, який рахує кількість чисел у масиві, які рівні 150, 1000 або 10000.

#include <stdio.h>

#include <stdlib.h>

int main()

{

system ("chcp 1251 & cls");

int numbers [14];

int count\_150 = 0, count\_1000 = 0, count\_10000 = 0;

printf ("Введіть 14 чисел:\n");

for (int i = 0; i < 14; i++) {

scanf ("%d", &numbers[i]);

}

for (int i = 0; i < 14; i++) {

if (numbers[i] == 150) {

count\_150++;

} else if (numbers[i] == 1000) {

count\_1000++;

} else if (numbers[i] == 10000) {

count\_10000++;

}

}

printf ("Кількість чисел = 150: %d\n", count\_150);

printf ("Кількість чисел = 1000: %d\n", count\_1000);

printf ("Кількість чисел = 10000: %d\n", count\_10000);

return 0;

}

**Результат:**

Введіть 14 чисел:10000 10000 10000 10000 10000 10000 10000 10000 10000 10000 10000 10000 10000 10000 Кількість чисел = 150: 0

Кількість чисел = 1000: 0

Кількість чисел = 10000: 14

**Висновок:**

Узадачі 7.2, програма призначена для підрахунку кількості введених користувачем чисел, які рівні 150, 1000 або 10000. Я використав масив numbers для зберігання введених значень, а потім цикл для перевірки кожного числа та підрахунку кількості чисел для кожного з трьох визначених значень. Код містить необхідні елементи для вирішення цієї задачі, і він вірно визначає кількість чисел, які рівні 150, 1000 або 10000.

У задачі 7.1, програма дозволяє користувачеві вводити речення, а потім перевіряє, чи містить введене речення слово "програма". Я використав функцію strstr для пошуку слова, а також конвертував всі символи у реченні в нижній регістр (tolower) для більш точного пошуку. Програма правильно визначає наявність чи відсутність слова "програма" у введеному реченні.

Я набув ґрунтовних вмінь і практичних навичок розроблення

циклічних алгоритмів розв’язування обчислювальних задач та їх формального

представлення мовою програмування С задля реалізації

програмних засобів у вільному кросплатформовому Code::Blocks IDE.