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

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 КІ-23

Попов О. М.

ПЕРЕВІРИВ

ст. викладач кафедри кібербезпеки   
Дрєєва Г.М.

Кропивницький – 2024

**Лабараторна робота №7**

**Тема:** ПРОГРАМНА РЕАЛІЗАЦІЯ ОБРОБЛЕННЯ МАСИВІВ ДАНИХ ТА СИМВОЛЬНОЇ ІНФОРМАЦІЇ ЗА СТАНДАРТОМ UNICODE

**МЕТА**: полягає у набутті ґрунтовних вмінь і практичних навичок синтезу алгоритмів оброблення масивів даних та символьної (текстової) інформації у кодуваннях UTF-8 і CP866, їх програмної реалізації мовою програмування мовою програмування С (ISO/IEC 9899:2018) задля реалізації програмних засобів у вільному кросплатформовому Code::Blocks IDE.

**Завдання:**

1. У веббраузері перейти за посиланням https://github.com/ та зареєструватись на вебсервісі системи контролю версій GitHub. 2. Після створення облікового запису (account) і підтвердження реєстрації через електронну пошту, у Personal settings заповнити Public profile: Name: власне ім’я й прізвище латиницею, Profile picture: офіційне фото (ділового стилю), URL: www.kntu.kr.ua, Company: Central Ukrainian National Technical University, Location: Ukraine, Kropyvnytskyi. 3. Продемонструвати викладачеві створений обліковий запис, після чого надіслати GitHub username на dorenskyiop@kntu.kr.ua. 4. Отримати у викладача завдання за варіантом, задокументувати мету лабораторної роботи, завдання, номер варіанту. 5. На одному з локальних дисків ПК (D:, E: чи ін.) створити теку BMTP-LAB7-прізвище, в ній ― теки prj, Software та TestSuite. 6. Виконати аналіз умови і постановку задачі 7.1, встановлених вимог до програмного засобу, який розроблятиметься, на основі чого розробити й задокументувати тест-сьют TS\_7\_1, який повинен складатись з не менш як десяти тест-кейсів. 7. Виконати процедурно-орієнтовану алгоритмізацію задачі 7.1 з врахуванням вимоги про програмне оброблення текстової інформації у UTF-8, при чому кодування вхідних і вихідних даних залежить від джерела й приймача інформації відповідно; здобутий алгоритм розв’язування задачі 7.1 задокументувати. 8. В Code::Blocks IDE створити новий проект консольного застосунку з назвою прізвище-task\_7\_1 у /prj. 9. Алгоритм розв’язування задачі 7.1 реалізувати мовою програмування С (С18), проект прізвище-task\_7\_1 закрити. 10. З теки проекта прізвище-task\_7\_1 за допомогою "Блокнот" відкрити файл main.c. 11. Перезберегти файл main.c у UTF-8 (Файл → Зберегти як → Кодування → UTF-8 → Зберегти). 12. Закрити файл main.c. 44 Базові методології та технології програмування 13. В Code::Blocks IDE відкрити прізвище-task\_7\_1, скомпілювати і закрити проект; створений у /prj файл програми прізвищеtask\_7\_1.ехе скопіювати у /BMTP-LAB7-прізвище/Software. 14. За допомогою тестового набору TS\_7\_1 із теки /TestSuite, виконати системне тестування прізвище-task\_7\_1.ехе, який знаходиться у теці /BMTP-LAB7-прізвище/Software; отриманий тестовий артефакт — заповнений тест-сьют TS\_7\_1 — зберегти та долучити до звіту як додаток. 15. За умови, що статус всіх тест-кейсів TS\_7\_1 є passed, вихідний код (лістинг) проекту прізвище-task\_7\_1 задокументувати; у іншому випадку слід виконати відлагодження запропонованого алгоритму і/або ПЗ, проект прізвище-task\_7\_1 скомпілювати, системне тестування додатка прізвище-task\_7\_1.ехе повторити, результати повторної реалізації й тестування задокументувати. 16. Виконати аналіз умови і постановку задачі 7.2, встановлених вимог до програмного засобу, який розроблятиметься, на основі чого розробити й задокументувати тест-сьют TS\_7\_2, який повинен складатись з не менш як десяти тест-кейсів. 17. Виконати процедурно-орієнтовану алгоритмізацію задачі 7.2, отриманий алгоритм розв’язування задачі 7.2 задокументувати. 18. В Code::Blocks IDE створити новий проект консольного застосунку з назвою прізвище-task\_7\_2 у /prj. 19. Алгоритм розв’язування задачі 7.2 реалізувати мовою програмування С (С18), скомпілювати проект; створений файл програми прізвище-task\_7\_2.ехе скопіювати у /Software. 20. За допомогою TS\_7\_2 із /TestSuite, виконати системне тестування /BMTP-LAB2-прізвище/Software/прізвище-task\_7\_2.ехе; заповнений TS\_7\_2 долучити до звіту як додаток. 21. За умови, що статус всіх тест-кейсів TS\_7\_2 є passed, вихідний код (лістинг) проекту прізвище-task\_7\_2 задокументувати; у іншому випадку слід виконати відлагодження запропонованого алгоритму і/або ПЗ, проект прізвище-task\_7\_2 скомпілювати, системне тестування додатка прізвище-task\_7\_2.ехе повторити, результати повторної реалізації й тестування задокументувати. 22. Одержані результати виконання завдань лабораторної роботи (/BMTP-LAB7-прізвище) зберегти на флеш-накопичувач, у хмарному сховищі даних тощо.23. Проаналізувати хід виконання завдань і одержані результати, сформулювати обґрунтовані висновки15 та викласти їх обсягом не менше двох сторінок машинного (комп’ютерного) тексту; також слід зазначити особисті враження від виконання лабораторної роботи, викласти вмотивовані пропозиції, обґрунтовані зауваження, конструктивну критику, рекомендації тощо. 24. Інсталювати Git (Git Bash) <https://git-scm.com/book/uk/v2/> ВступІнсталяція-Git, використовуючи “Pro Git” https://gitscm.com/book/uk/v2. 25. На github.com/student-name створити Git-репозиторій (New → Create a new repository) та за допомогою Git (Git Bash) завантажити у нього теку BMTP-LAB7-прізвище (git init, git add, git commit, git branch, git remote, git push; https://techglimpse.com/git-push-github-token-based-passwordless/), а також звіт, оформлений згідно з ДСТУ 3008:2015 зі структурою й змістом, визначеними цим порядком виконання роботи. 26. Додати викладача і одногрупників до спільної роботи над проєктом BMTP-LAB7-прізвище: Settings → Access → Collaborators → Add people → Add NAME to REPOSITORY (NAME викладача — на дошці в аудиторії, у якій проходить заняття). 27. У комітах Git-репозиторію BMTP-LAB7-прізвище (https://github.com/ . . . /commits/main/) відреагувати і/або відповісти на коментарі співучасників ІТ-проєкту ( Commits), захистити результати й здобути зарахування роботи викладачем.

**Варіант №9**

**Задача 7.1:**

Користувач вводить слово (рядок кириличних символів без проміжків), яке закінчується “%”, “~” або “\*”. Вивести кількість складів “ма” у введеному слові; якщо зазначений склад відстуній. Вивести всі приголосні з уведеного слова.

**Лістинг 7.1**

#include <stdio.h>

#include <stdlib.h>

#include <locale.h>

#include <string.h>

#include <ctype.h>

int main() {

setlocale(LC\_ALL, "");

char word[100];

int count = 0;

char golosni[] = "ауеоюяіи";

printf("Введіть слово яке закінчується на %% , ~ або \*\n");

scanf("%s", word);

char \*substr = word;

while ((substr = strstr(substr, "ма")) != NULL) {

count++;

substr++;

}

if (count == 0) {

for (int i = 0; word[i] != '\0'; i++) {

char lower = tolower(word[i]);

for (int j = 0; golosni[j] != '\0'; j++) {

if (lower == golosni[j]) {

printf("%c", word[i]);

}

}

}

printf("\n");

}

printf("Кількість mа: %d\n", count);

return 0;

}

|  |  |
| --- | --- |
| Назва тестового набору  Test Suite Description | TS\_lab7 |
| Назва проекта / ПЗ  Name of Project / Software | Gonchar-task\_7\_1 |
| Рівень тестування  Level of Testing | системний / System Testing |
| Автор тест-сьюта  Test Suite Author | Попов Максим |
| Виконавець  Implementer | Попов Максим |

|  |  |  |  |
| --- | --- | --- | --- |
| 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) |
| ТС-1 | 1. Введення омаома~ | Введ?ть слово яке зак?нчується на , ~ або \*  К?льк?сть ма: 2 | passed |
| ТС-2 | 1. Введення амам\* | Введ?ть слово яке зак?нчується на , ~ або \*  aa  К?льк?сть ма: 0 | passed |
| ТС-3 | 1. Введення мамама% | Введ?ть слово яке зак?нчується на , ~ або \*  К?льк?сть ма: 3 | passed |
| ТС-4 | 1. Введення ма~ | Введ?ть слово яке зак?нчується на , ~ або \*  К?льк?сть ма: 1 | passed |
| ТС-5 | 1. Введення аамюю\* | Введ?ть слово яке зак?нчується на , ~ або \*  аааюю  К?льк?сть ма: 0 | passed |
| ТС-6 | 1. Введення Доренський~ | Введ?ть слово яке зак?нчується на , ~ або \*  оеи  К?льк?сть ма: 0 | passed |
| ТС-7 | 1. Введення ммммм\* | Введ?ть слово яке зак?нчується на , ~ або \*  К?льк?сть ма: 0 | passed |
| ТС-8 | 1. Введення aaaaaaaaaa% | Введ?ть слово яке зак?нчується на , ~ або \*  aaaaaaaaaa  К?льк?сть ма: 0 | passed |
| ТС-9 | 1. Введення матттттма~ | Введ?ть слово яке зак?нчується на , ~ або \*  К?льк?сть ма: 2 | passed |
| ТС-10 | 1. Введення ддлдоддду\* | Введ?ть слово яке зак?нчується на , ~ або \*  оу  К?льк?сть ма: 0 | passed |

**Задача 7.2:**Вхід: масив з 15 натуральних чисел.

Вихід: сума та різниця найбільшого і найменшого елементів масиву.

**Лістинг 7.2**

#include <stdio.h>

#include <stdlib.h>

#include <locale.h>

int main() {

char \*locale = setlocale(LC\_ALL, "");

int numbers[15];

int max, min, rizn, sum;

printf("Введіть 15 натуральних чисел:\n");

for (int i = 0; i < 15; i++) {

printf("Число %d: ", i + 1);

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

}

max = numbers[0];

min = numbers[0];

for (int i = 1; i < 15; i++) {

if (numbers[i] > max) {

max = numbers[i];

}

if (numbers[i] < min) {

min = numbers[i];

}

}

rizn = max - min;

sum = max + min;

printf("Сума найбільшого і найменшого числа: %d\n", sum);

printf("Різниця між найбільшим та найменшим числом: %d\n", rizn);

return 0;

}

|  |  |
| --- | --- |
| Назва тестового набору  Test Suite Description | TS\_lab7\_2 |
| Назва проекта / ПЗ  Name of Project / Software | Gonchar-task\_7\_2 |
| Рівень тестування  Level of Testing | системний / System Testing |
| Автор тест-сьюта  Test Suite Author | Попов Максим |
| Виконавець  Implementer | Попов Максим |

|  |  |  |  |
| --- | --- | --- | --- |
| 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) |
| ТС-1 | 1. Введення 1 2. Введення 5 3. Введення 1 4. Введення 2 5. Введення 5 6. Введення 10 7. Введення -10 8. Введення 2 9. Введення 3 10. Введення 5 11. Введення 1 12. Введення 2 13. Введення 45 14. Введення 1 15. Введення 2 | Введ?ть 15 натуральних чисел:  Число 1: 1  Число 2: 5  Число 3: 1  Число 4: 2  Число 5: 5  Число 6: 10  Число 7: -10  Число 8: 2  Число 9: 3  Число 10: 5  Число 11: 1  Число 12: 2  Число 13: 45  Число 14: 1  Число 15: 2  Сума найб?льшого ? найменшого числа: 35  Р?зниця м?ж найб?льшим та найменшим числом: 55 | passed |
| ТС-2 | 1. Введення 0 2. Введення 0 3. Введення 0 4. Введення 0 5. Введення 0 6. Введення 0 7. Введення 0 8. Введення 0 9. Введення 0 10. Введення 0 11. Введення 0 12. Введення 0 13. Введення 0 14. Введення 0 15. Введення 0 | Введ?ть 15 натуральних чисел:  Число 1: 0  Число 2: 0  Число 3: 0  Число 4: 0  Число 5: 0  Число 6: 0  Число 7: 0  Число 8: 0  Число 9: 0  Число 10: 0  Число 11: 0  Число 12: 0  Число 13: 0  Число 14: 0  Число 15: 0  Сума найб?льшого ? найменшого числа: 0  Р?зниця м?ж найб?льшим та найменшим числом: 0 | passed |
| ТС-3 | 1. Введення 100 2. Введення -121 3. Введення 1 4. Введення 5 5. Введення 293 6. Введення 18 7. Введення 59 8. Введення 28 9. Введення 3 10. Введення 1 11. Введення 6 12. Введення 1 13. Введення 1 14. Введення 1 15. Введення 1 | Введ?ть 15 натуральних чисел:  Число 1: 100  Число 2: -121  Число 3: 1  Число 4: 5  Число 5: 293  Число 6: 18  Число 7: 59  Число 8: 28  Число 9: 3  Число 10: 1  Число 11: 6  Число 12: 1  Число 13: 1  Число 14: 1  Число 15: 1  Сума найб?льшого ? найменшого числа: 172  Р?зниця м?ж найб?льшим та найменшим числом: 414 | passed |
| ТС-4 | 1. Введення 12391 2. Введення -12391 3. Введення 0 4. Введення 12 5. Введення 3 6. Введення 0 7. Введення 5 8. Введення 12 9. Введення 1 10. Введення 512 11. Введення 2 12. Введення 5 13. Введення 1 14. Введення 5 15. Введення 2 | Введ?ть 15 натуральних чисел:  Число 1: 12391  Число 2: -12391  Число 3: 0  Число 4: 12  Число 5: 3  Число 6: 0  Число 7: 5  Число 8: 12  Число 9: 1  Число 10: 512  Число 11: 2  Число 12: 5  Число 13: 1  Число 14: 5  Число 15: 2  Сума найб?льшого ? найменшого числа: 0  Р?зниця м?ж найб?льшим та найменшим числом: 24782 | passed |
| ТС-5 | 1. Введення 0 2. Введення 1 3. Введення 1 4. Введення 1 5. Введення 1 6. Введення 11 7. Введення 1 8. Введення 1 9. Введення 1 10. Введення 1 11. Введення 1 12. Введення 1 13. Введення 1 14. Введення 1 15. Введення 1 | Введ?ть 15 натуральних чисел:  Число 1: 0  Число 2: 1  Число 3: 1  Число 4: 1  Число 5: 1  Число 6: 11  Число 7: 1  Число 8: 1  Число 9: 1  Число 10: 1  Число 11: 1  Число 12: 1  Число 13: 1  Число 14: 1  Число 15: 1  Сума найб?льшого ? найменшого числа: 11  Р?зниця м?ж найб?льшим та найменшим числом: 11 | passed |
| ТС-6 | 1. Введення 10 2. Введення 10 3. Введення 101 4. Введення 10 5. Введення 101 6. Введення 12 7. Введення 12 8. Введення 1 9. Введення 1 10. Введення 1 11. Введення 1 12. Введення 1 13. Введення 1 14. Введення 1 15. Введення 1 | Введ?ть 15 натуральних чисел:  Число 1: 10  Число 2: 10  Число 3: 101  Число 4: 10  Число 5: 101  Число 6: 12  Число 7: 12  Число 8: 1  Число 9: 1  Число 10: 1  Число 11: 1  Число 12: 1  Число 13: 1  Число 14: 1  Число 15: 1  Сума найб?льшого ? найменшого числа: 102  Р?зниця м?ж найб?льшим та найменшим числом: 100 | passed |
| ТС-7 | 1. Введення -1 2. Введення -1 3. Введення -2 4. Введення -3 5. Введення -4 6. Введення -5 7. Введення -6 8. Введення -7 9. Введення -8 10. Введення -1 11. Введення -1 12. Введення -1 13. Введення -1 14. Введення -6 15. Введення -2 | Введ?ть 15 натуральних чисел:  Число 1: -1  Число 2: -1  Число 3: -2  Число 4: -3  Число 5: -4  Число 6: -5  Число 7: -6  Число 8: -7  Число 9: -8  Число 10: -1  Число 11: -1  Число 12: -1  Число 13: -5  Число 14: -6  Число 15: -2  Сума найб?льшого ? найменшого числа: -9  Р?зниця м?ж найб?льшим та найменшим числом: 7 | passed |
| ТС-8 | 1. Введення 1 2. Введення 2 3. Введення 3 4. Введення 4 5. Введення 5 6. Введення 6 7. Введення 7 8. Введення 8 9. Введення 9 10. Введення 10 11. Введення 11 12. Введення 12 13. Введення 13 14. Введення 14 15. Введення 15 | Введ?ть 15 натуральних чисел:  Число 1: 1  Число 2: 2  Число 3: 3  Число 4: 4  Число 5: 5  Число 6: 6  Число 7: 7  Число 8: 8  Число 9: 9  Число 10: 10  Число 11: 11  Число 12: 12  Число 13: 13  Число 14: 14  Число 15: 15  Сума найб?льшого ? найменшого числа: 16  Р?зниця м?ж найб?льшим та найменшим числом: 14 | passed |
| ТС-9 | 1. Введення -1 2. Введення -2 3. Введення -3 4. Введення -4 5. Введення -5 6. Введення -6 7. Введення -7 8. Введення -8 9. Введення -9 10. Введення -10 11. Введення -11 12. Введення -12 13. Введення -13 14. Введення -14 15. Введення -15 | Введ?ть 15 натуральних чисел:  Число 1: -1  Число 2: -2  Число 3: -3  Число 4: -4  Число 5: -5  Число 6: -6  Число 7: -7  Число 8: -8  Число 9: -9  Число 10: -10  Число 11: -11  Число 12: -12  Число 13: -13  Число 14: -14  Число 15: -15  Сума найб?льшого ? найменшого числа: -16  Р?зниця м?ж найб?льшим та найменшим числом: 14 | passed |
| ТС-10 | 1. Введення 100100 2. Введення -100100 3. Введення 0 4. Введення 0 5. Введення 0 6. Введення 0 7. Введення 0 8. Введення 0 9. Введення 0 10. Введення 0 11. Введення 0 12. Введення 0 13. Введення 0 14. Введення 0 15. Введення 1 | Введ?ть 15 натуральних чисел:  Число 1: 100100  Число 2: -100100  Число 3: 0  Число 4: 0  Число 5: 0  Число 6: 0  Число 7: 0  Число 8: 0  Число 9: 0  Число 10: 0  Число 11: 0  Число 12: 0  Число 13: 0  Число 14: 0  Число 15: 1  Сума найб?льшого ? найменшого числа: 0  Р?зниця м?ж найб?льшим та найменшим числом: 200200 | passed |

**Висновок:** Виконання цієї лабораторної роботи (точніше завдання 7.1) змусила мене провести декілька годин на StackOverflow . У процесі пошуку інформації для виконання завдання, я заволодів новими знаннями, що дало змогу переробити алгоритм для завдання 7.1.

Я задоволений, що в подальшому ми будемо використовувати Git, а також вивчати C++.

**Відповіді на питання:**

1. Функція `git init` створює новий репозиторій Git. Вона ініціалізує порожній репозиторій або перетворює існуючий каталог в репозиторій Git.

Функція `git add` додає зміни до індексу для подальшого коміту. `git add .` додає всі змінені, видалені та нові файли у поточному каталозі та всі його підкаталоги.

Функція `git commit` фіксує стан індексу в репозиторії. Команда `git commit -m "текст\_коміту"` робить коміт з повідомленням про коміт, яке надається у рядку.

Функція `git remote add` додає новий віддалений репозиторій. Наприклад, `git remote add origin git@github.com:username/BMTP-LAB7-прізвище.git` додає віддалений репозиторій з назвою "origin" і URL-адресою git@github.com:username/BMTP-LAB7-прізвище.git.

Функція `git push` надсилає зміни з локального репозиторію на віддалений сервер.

Наприклад, `git push https://<GITHUB\_ACCESS\_TOKEN>@github.com/<GITHUB\_USERNAME>/<REPOSITORY\_NAME>.git` надсилає зміни на GitHub за допомогою HTTPS-з'єднання та токена доступу GitHub.