**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ**

**НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ "ЛЬВІВСЬКА ПОЛІТЕХНІКА"**

Інститут **ІКНІ**

Кафедра **ПЗ**

**ЗВІТ**

До лабораторної роботи № 4

**На тему:** *“* Робота з масивами в С*”*

**З дисципліни:** *“Основи програмування”*

**Лектор:**

ст.викл. каф. ПЗ

Муха Т. О.

**Виконав:**

гр. ПЗ - 18

Таранець Є. А.

**Прийняв:**

ст.викл. каф. ПЗ

Кутельмах Р.К.

« \_\_\_\_ » \_\_\_\_\_\_\_\_ 2022 р.

∑= \_\_\_\_ .

Львів – 2022

**Тема роботи:** Символьні рядки в С.

**Мета роботи:** Здобути практичні навики опрацювання текстової інформації з

врахуванням особливостей організації символьних рядків у мові С. Вивчити основні засоби потокового вводу/виводу в С.

**Лабораторне завдання**

1. Ознайомитися з теоретичним матеріалом викладеним вище в даній інструкції і виконати

приклади програм.

2. Одержати індивідуальне завдання з Додатку 1.

3. Розробити алгоритм розв’язання індивідуального завдання і подати його у вигляді блоксхеми.

4. Скласти програму на мові С у відповідності з розробленим алгоритмом.

5. Виконати обчислення по програмі.

6. Одержати індивідуальне завдання з Додатку 2.

7. Розробити алгоритм розв’язання індивідуального завдання і подати його у вигляді блоксхеми.

8. Скласти програму на мові С у відповідності з розробленим алгоритмом.

9. Виконати обчислення по програмі при різних значеннях точності і порівняти отримані

результати.

10.Підготувати та здати звіт про виконання лабораторної роботи

**Індивідуальне завдання**

**Завдання 1:** Надрукувати всі слова з введеного речення, які містять понад 8 літер.

**Завдання 2:** Скласти програму на мові С, яка обчислюватиме значення виразу з лабораторної роботи

№1 Додаток 1 (варіант 17).

**Теоретичні відомості**

Рядкові константи (літерали) в мові С записуються як послідовність довільних символів  
взятих у подвійні лапки: "...". В оперативній пам'яті їм виділяється ділянка, обсяг якої на один  
байт більший за кількість символів у рядку. В цей додатковий байт автоматично записується  
нуль-символ (‘\0’), який надалі слугуватиме ознакою кінця рядка.

Процеси опрацювання символьних рядків базуються на двох основних властивостях  
рядків:  
1) ім'я символьного рядка є константним вказівником на його перший символ;  
2) кінець рядка задається нуль-символом ' \0'.

Для звертання до символів рядка застосовують як індексну, так і вказівникову систему  
доступу до елементів масиву.

***Символьні рядки, які опрацьовуються функціями <string.h>, мають обов'язково  
закінчуватись ' \0'***. У разі звертання до функцій конкатенації (об'єднання) та копіювання  
рядків треба забезпечити, щоб розмірність масиву символів sr була достатньою для запису  
рядка результату, оскільки функції не контролюють довжин рядків. Крім цього, рядки sr і s у функціях конкатенації та копіювання не повинні перекриватись.

Функція **printf ()** призначена для форматованого виведення. Вона переводить дані в  
символьне представлення і записує отриману послідовність символів в стандартний потік  
виводу, що приводить, як правило, до виводу символів на екран. При цьому у програміста є  
можливість форматувати дані, тобто впливати на їх виведення на екран.

%d або %i – ціле число типу **int** зі знаком в десятковій системі числення;

|  |  |
| --- | --- |
| %u  %x  %X  %o  %hd  %hu  %hx  %ld  %lu  %lld  %llu  %lx  %llx  %f  %F  %lf  %e  %g  %c  %s  %p  %n  %% | – ціле число типу **unsigned int**; – ціле число типу **unsigned int** зі знаком в шістнадцятковій системі числення; – ціле число типу **unsigned int** зі знаком в шістнадцятковій системі числення (прописом); – ціле число типу **unsigned int** зі знаком в вісімковій системі числення; – ціле число типу **short** зі знаком в десятковій системі числення; – ціле число типу **unsigned short**; – ціле число типу **short** зі знаком в шістнадцятковій системі числення; – ціле число типу **long int** зі знаком в десятковій системі числення; – ціле число типу **unsigned long int**; – ціле число типу **long long int** зі знаком в десятковій системі числення; – ціле число типу **unsigned long long int**; – ціле число типу **long int** зі знаком в шістнадцятковій системі числення; – ціле число типу **long long int** зі знаком в шістнадцятковій системі числення; – дійсне число (число з плаваючою крапкою типу **float**); – дійсне число (число з плаваючою крапкою типу **float**) (прописом); – дійсне число подвійної точності (число з плаваючою крапкою типу **double**);  – дійсне число в експоненційній формі (числа з плаваючою крапкою типу **float** в експоненційній формі);  – дійсне число (автоматичний вибір серед коротшого %f чи %e) – один символ (об’єкт типу **char**); – набір символів з останнім нульовим символом '\0' (нуль-термінальний рядок). – вказівник (об’єкт типу **void\***); – нічого не виводиться. Відповідний аргумент (об’єкт) повинен мати тип **int**; – символ % |

**Хід роботи**

**№1**

**Код програми:**

**Назва файлу: Lab3\_1.c**

##include <stdio.h>  
#include <stdlib.h>  
#include <string.h>  
#include <windows.h>  
  
int main() {  
 char str[500];  
 char \*limits = " ,.-",\*w;  
 printf("Enter your sentence: \n");  
 gets(str);  
 if (str[strlen(str) - 1] == '.') {  
 w = strtok(str, limits);  
 while (w != NULL) {  
 if (strlen(w) > 8) {  
 printf("%s\n", w);  
 }  
 w = strtok(NULL, limits);  
 }  
 }  
 else {  
 printf("Error");  
 }  
 return 0;  
 }

**Протокол роботи:**

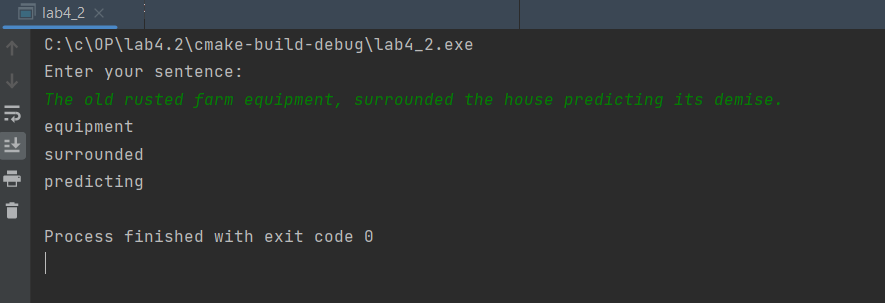


Рисунок 1. Приклад виведення розв'язку першого завдання

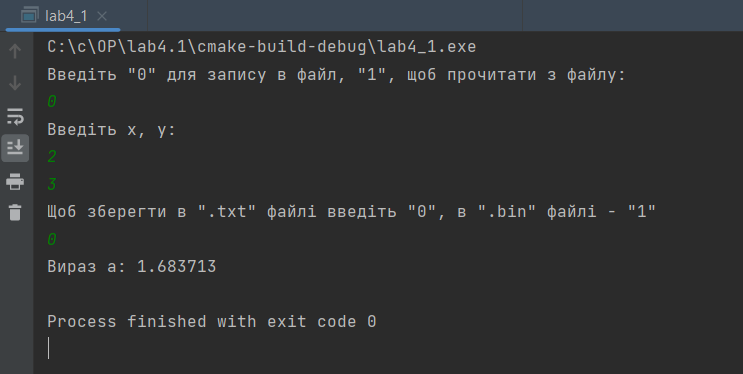
**№2**

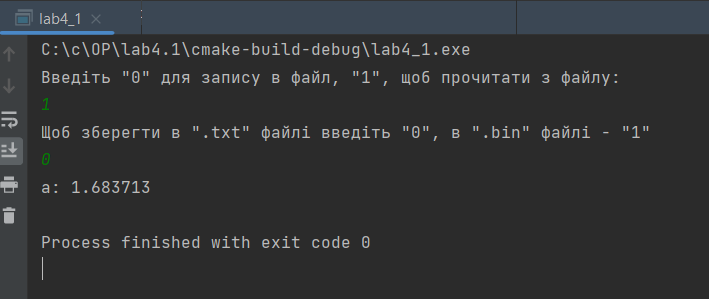
**Код програми:**

**Назва файлу: Lab3\_2.c**

#include <stdio.h>  
#include <math.h>  
#include <time.h>  
#include <windows.h>  
void gettime(FILE\* log);  
int main()  
{  
 SetConsoleOutputCP(CP\_UTF8);  
 FILE\* log = fopen("log.txt", "a");  
 gettime(log);  
 fprintf(log, "Програму запущено\n");  
 int b,c;  
 double a, x, y, k[2];  
 printf("Введіть \"0\" для запису в файл, \"1\", щоб прочитати з файлу:\n");  
 scanf("%d", &b);  
 if (b == 0) {  
 FILE\* in = fopen("in.bin", "wb");  
 printf("Введіть x, y:\n");  
 scanf("%lf%lf", &x, &y);  
 gettime(log);  
 fprintf(log, "Значення x = %lf, y = %lf введено користувачем\n", x,y);  
 if (in != 0)  
 {  
 k[0] = x;  
 k[1] = y;  
 fwrite(k, sizeof(k), 1, in);  
 fclose(in);  
 }  
 fopen("in.bin", "rb");  
 fread(k, sizeof(k), 1, in);  
 x = k[0];  
 y = k[1];  
 a = fabs(pow(x, y/x) - pow(y/x, 1.0/3.0));  
 printf("Щоб зберегти в \".txt\" файлі введіть \"0\", в \".bin\" файлі - \"1\"\n");  
 scanf\_s("%d", &c);  
 if (c == 0)  
 {  
 FILE\* out = fopen("out.txt", "w");  
 printf("Вираз a: %lf\n", a);  
 fprintf(out, "Вираз a: %lf\n", a);  
 fclose(out);  
 gettime(log);  
 fprintf(log, "Файл збережено в \"C:\\c\\OP\\lab4.1\\cmake-build-debug\\out.txt\"\n");  
 }  
 if (c == 1)  
 {  
 FILE\* out = fopen("out.bin", "wb");  
 fwrite(&a, sizeof(a), 1, out);  
 printf("Вираз a: %lf\n", a);  
 fclose(out);  
 gettime(log);  
 fprintf(log, "Файл збережено в \"C:\\c\\OP\\lab4.1\\cmake-build-debug\\out.bin\"\n");  
 }  
 }  
 else if (b == 1) {  
 FILE\* in = fopen("in.bin", "rb");  
 fread(k, sizeof(k), 1, in);  
 x = k[0];  
 y = k[1];  
 gettime(log);  
 fprintf(log, "Значення x = %lf, y = %lf,взято з файлу \"C:\\c\\OP\\lab4.1\\cmake-build-debug\\in.bin\"\n", x, y);  
 a = fabs(pow(x, y/x) - pow(y/x, 1.0/3.0));  
 gettime(log);  
 printf("Щоб зберегти в \".txt\" файлі введіть \"0\", в \".bin\" файлі - \"1\"\n");  
 scanf\_s("%d", &c);  
 fprintf(log, "Вираз a: %lf\n", a);  
 if (c == 0)  
 {  
 FILE\* out\_file = fopen("out.txt", "w");  
 fprintf(out\_file, "a: %lf\n", a);  
 printf("a: %lf\n", a);  
 gettime(log);  
 fprintf(log, "Файл збережено в \"C:\\c\\OP\\lab4.1\\cmake-build-debug\\out.txt\"\n");  
  
 }  
 if (c == 1)  
 {  
 FILE\* out\_file = fopen("out.bin", "wb");  
 fwrite(&a, sizeof(a), 1, out\_file);  
 printf("a: %lf\n", a);  
 gettime(log);  
 fprintf(log, "Файл збережено в \"Файл збережено в \"C:\\c\\OP\\lab4.1\\cmake-build-debug\\out.bin\"\n");  
 }  
 }  
 else  
 {  
 gettime(log);  
 fprintf(log, "Error, введено неправильне значення\n");  
 printf("Error, введено неправильне значення");  
 }  
 gettime(log);  
 fprintf(log, "Програму виконано\n");  
 fclose(log);  
}  
  
  
void gettime(FILE\* log) {  
 time\_t t;  
 time(&t);  
 char time[26];  
 ctime\_s(time, sizeof time, &t);  
 time[24] = ']';  
 time[25] = '\0';  
 fprintf(log, "[%s\n", time);  
}

**Протокол роботи:**





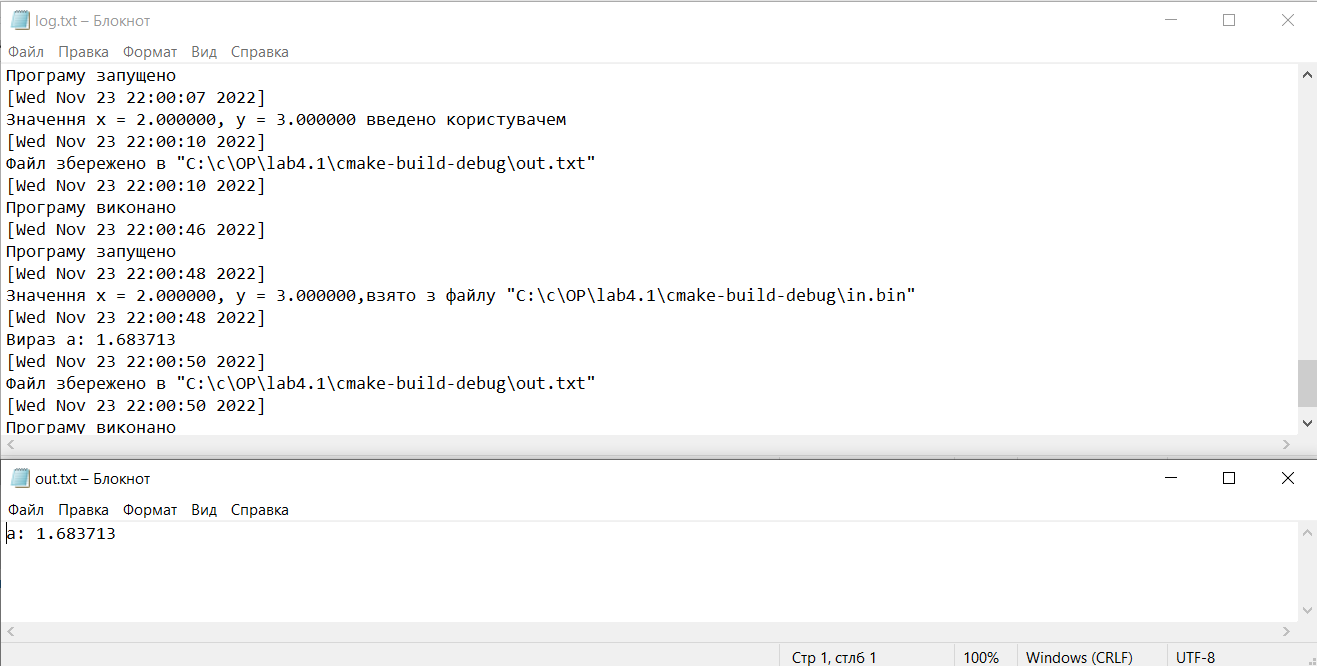


Рисунок 4. Приклад виведення розв'язку другого завдання

**Висновок:** На цій лабораторній роботі я здобув практичні навики опрацювання текстової інформації з

врахуванням особливостей організації символьних рядків у мові С, вивчив основні засоби

потокового вводу/виводу в С.