

Міністерство освіти і науки України
Національний університет “Львівська політехніка”
Інститут комп’ютерних наук та інформаційних технологій
Кафедра програмного забезпечення



Звіт

Про виконання лабораторної роботи №4

на тему:

«Символьні рядки в С. Ввід та вивід»

з дисципліни «Основи програмування»

Лектор:

ст. викл. каф. ПЗ

Муха Т.О.

Виконав:

ст. гр. ПЗ-11

Ясногородський Н.В.

Прийняв:

асист. каф. ПЗ

Дивак І.В.

« __ » _____ 2021 р.

Σ = _____ .

Львів – 2021

Тема: Символьні рядки в С. Ввід та вивід

Мета: Здобути практичні навички опрацювання текстової інформації з урахуванням особливостей організації символьних рядків у мові С. Вивчити основні засоби потокового вводу/виводу в С

ЗАВДАННЯ

Завдання 1.

Вважаючи, що введене речення з клавіатури складається з довільної кількості слів, між якими є довільна кількість пробілів, і закінчується речення крапкою, написати програму для розв'язання завдання:

4. Визначити відсоток вживання у введеному реченні кожної з голосних літер. Результат записати в спадному порядку.

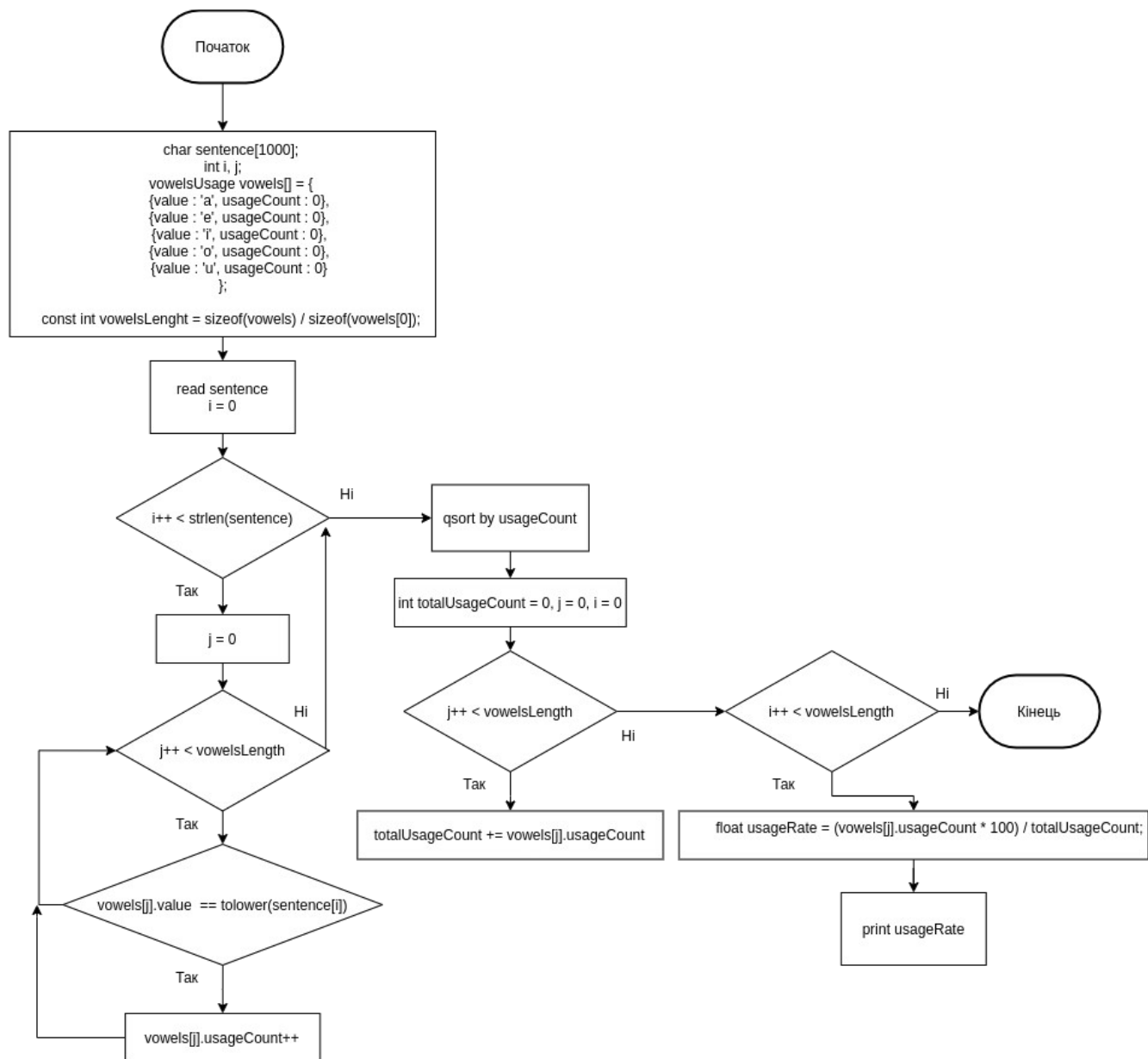
Завдання 2.

Скласти програму на мові С, яка обчислюватиме значення виразу з лабораторної роботи №1 Додаток 1 “Обчислення заданих арифметичних виразів”. Значення параметрів X, Y, Z прочитати із заданого бінарного файлу (вводиться користувачем з клавіатури). Результат записати у файл, теж вказаний користувачем. Формат файлу (бінарний чи текстовий) також задається користувачем. При цьому робота програми повинна бути запротокольована. Виконання основних подій, починаючи зі старту програми, має бути відображене у log-файлі з часовими мітками (розміщення вибрати самостійно). Старі записи у log-файлі мають зберігатися. Перелік подій для логування:

- старт програми
- відкриття файлу з параметрами
- обчислення значення виразу
- запис обчисленого значення у вихідний файл
- завершення програми

Блок схеми

Завдання 1



ТЕКСТ ПРОГРАМИ

Завдання 1

```
#include <stdio.h>
#include <math.h>
#include <string.h>
#include <stdlib.h>

You, a month ago | 1 author (You)
typedef struct Vowels
{
    char value;
    int usageCount;
} vowelsUsage;

int vowelsUsageComparator(const void *a, const void *b)
{
    vowelsUsage *ia = (vowelsUsage *)a;
    vowelsUsage *ib = (vowelsUsage *)b;
    return ib->usageCount - ia->usageCount;
}

int main(void)
{
    printf("Task 4\n");

    printf("Enter sentence to perform analysis. Sentence must end with '\\.\\' : ");
    char sentence[1000];
    scanf("%[^\n]s", sentence);

    vowelsUsage vowels[] = {
        {value : 'a', usageCount : 0},
        {value : 'e', usageCount : 0},
        {value : 'i', usageCount : 0},
        {value : 'o', usageCount : 0},
        {value : 'u', usageCount : 0},
    };

    const int vowelsLenght = sizeof(vowels) / sizeof(vowels[0]);
    const int sentenceLength = strlen(sentence);

    for (int i = 0; i < sentenceLength; i++)
    {
        for (int j = 0; j < vowelsLenght; j++)
        {
            if (vowels[j].value == tolower(sentence[i]))
            {
                vowels[j].usageCount++;
            }
        }
    }

    qsort(vowels, vowelsLenght, sizeof(vowels[0]), vowelsUsageComparator);

    int totalUsageCount = 0;

    for (int j = 0; j < vowelsLenght; j++)
    {
        totalUsageCount += vowels[j].usageCount;
    }

    if (!totalUsageCount)
    {
        printf("No vowels found! Exiting...\n");
        return -1;
    }

    for (int j = 0; j < vowelsLenght; j++)
    {
        float usageRate = (vowels[j].usageCount * 100) / totalUsageCount;
        printf("Vowel: %c, has %.2f%% usage rate\n", vowels[j].value, usageRate);
    }

    return 0;
}
```

Завдання 2

```

#include <stdio.h>
#include <time.h>
#include <stdarg.h>
#include <string.h>
#include "../1/1.h"

void logWithTimeStampW(const char *format, ...) __attribute__((format(printf, 1, 2)));

void logWithTimeStamp(const char *format, ...)
{
    char *string;
    va_list args;

    va_start(args, format);
    if (0 > vasprintf(&string, format, args))
        string = NULL;
    va_end(args);

    if (string)
    {
        _logWithTimeStamp(string);
        free(string);
    }
    else
    {
        _logWithTimeStamp("Error while logging a message: Memory allocation failed.\n");
    }
}

void _logWithTimeStamp(char str[])
{
    time_t t = time(NULL);
    struct tm *timestamp = localtime(&t);

    FILE *file = fopen("log.txt", "a");

    fprintf(file, "[%02d.%02d.%04d %02d:%02d:%02d] ",
        timestamp->tm_wday,
        timestamp->tm_mon + 1, // You, 4 days ago + feat: complete first part of 4th
        timestamp->tm_year + 1900,
        timestamp->tm_hour,
        timestamp->tm_min,
        timestamp->tm_sec);

    fprintf(file, str);
    fprintf(file, "\n");

    fclose(file);
}

int main(void)
{
    printf("Task from lab 1, addition 1\n");
    logWithTimeStamp("Program started.");

    char argsFile[60];
    printf("Enter binary file containing args, 60 chars max: ");
    scanf("%s", argsFile);

    FILE *file = fopen(argsFile, "rb");
    int x;
    fread(&x, sizeof(x), 1, file);
    logWithTimeStamp("Parameters file \"%s\" opened. x = %d.", argsFile, x);
    fclose(file);

    char outputFile[60];
    char outputFormat;
    printf("Enter output file name (without extension), 60 chars max: ");
    scanf("%s", outputFile);
    printf("Pick output format (b_binary/t_ext) : ");
    scanf(" %c", &outputFormat);

    double calcResult = firstCalcTask(x);
    logWithTimeStamp("Expression calculated. Result = %lf.", calcResult);

    switch (outputFormat)
    {
        case 'b':
            sprintf(outputFile, "%s.%s", outputFile, "bin");
            file = fopen(outputFile, "wb");
            fwrite(&calcResult, sizeof(double), 1, file);
            break;
        case 't':
            sprintf(outputFile, "%s.%s", outputFile, "txt");
            file = fopen(outputFile, "w");
            fprintf(file, "%lf", calcResult);
            break;
        default:
            break;
    }

    logWithTimeStamp("Output file \"%s\" saved.", outputFile);
    fclose(file);
    logWithTimeStamp("Program ended.\n");
    return 0;
}

```

РЕЗУЛЬТАТИ

```
[keep-simple@pc ~d/c/labs/1_semester/4]$ ./1
Task 4
Enter sentence to perform analysis. Sentence must end with "." : Hello, my name is Nikita! I am testing lab.
Vowel: i, has 38.00% usage rate
Vowel: a, has 30.00% usage rate
Vowel: e, has 23.00% usage rate
Vowel: o, has 7.00% usage rate
Vowel: u, has 0.00% usage rate
```

Рис 1. Результат виконання програми №1

```
Task from lab 1, addition 1
Enter binary file containing args, 60 chars max: input.bin
Enter output file name (without extension), 60 chars max: output
Pick output format (b_inary/t_ext) : t
[keep-simple@pc ~d/c/labs/1_semester/4]$
```

```
output.txt x
1_semester > 4 > output.txt
1 -2.345437
```

```
log.txt x
1_semester > 4 > log.txt
12 [04.09.2021 21:36:52] Program started.
11 [04.09.2021 21:36:59] Parameters file "input.bin" opened. x = 11.
10 [04.09.2021 21:37:23] Expression calculated. Result = -2.345437.
9 [04.09.2021 21:37:23] Output file "answers.txt" saved.
8 [04.09.2021 21:37:23] Program ended.
7
6 [08.09.2021 13:01:50] Program started.
5 [08.09.2021 13:01:54] Parameters file "input.bin" opened. x = 11.
4 [08.09.2021 13:02:13] Expression calculated. Result = -2.345437.
3 [08.09.2021 13:02:13] Output file "output.txt" saved.
2 [08.09.2021 13:02:13] Program ended.
1
13
```

Рис 2,3,4. Результат виконання програми №2

ВИСНОВКИ

Здобуто практичні навички опрацювання текстової інформації з урахуванням особливостей організації символічних рядків у мові C, вивчено основні засоби потокового вводу/виводу в C