Національний технічний університет України

«Київський політехнічний інститут імені Ігоря Сікорського»

**Алгоритмізація та програмування 1: Базові концепції програмування**

ЗВІТ

до лабораторної роботи №1

«**Файли: створення, запис, читання**»

*(ТЕМА)*

Варіант № 1

Дата «30» березня 2022 Виконав: студент 1 курсу

гр. ТР-15

Руденко Владислав Ігорович

*(П.І.Б.)*

Оцінка «\_\_\_\_\_\_\_\_\_» Перевірив:

Крячок Олександр Степанович

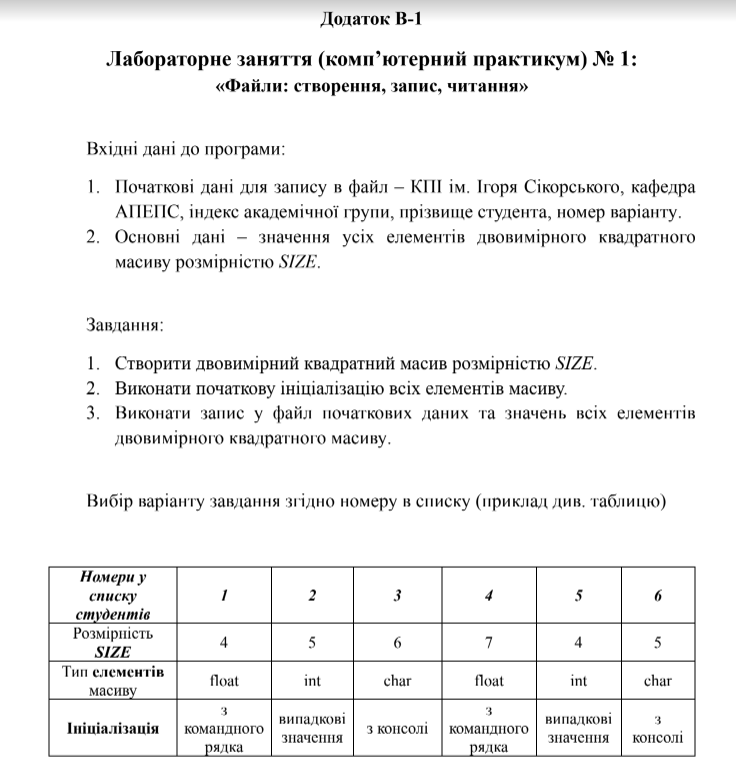
Дата «\_\_\_» \_\_\_\_\_\_\_ 20\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

*(П.І.Б., підпис)*

Київ – 2022

***Завдання до роботи:***

1. Ознайомитись з основними функціями для роботи з файлами.  
   2.     У якості індивідуального завдання необхідно написати програмний код, що реалізує *створення двовимірного квадратного масиву розмірністю SIZE; виконання початкової ініціалізації всіх елементів масиву; запис у файл****початкових даних****і****значень всіх елементів****двовимірного квадратного масиву; створює****копію****цього файлу з новим розширенням \*.bak.*  
   3.     Роздрукувати (вивести на екран) протокол виконання програмного коду.  
   4.     Вихідні дані вибрати за індивідуальними варіантами (див. Додаток B-1).



Додаток В- 1

***Теоритичні відомості:***

Потік – це джерело або одержувач даних; його можна пов'язати з диском або якимось іншим зовнішнім пристроєм. Бібліотека <stdio.h> підтримує два види потоків: текстовий та бінарний, хоча на деяких системах, зокрема у UNIXe, де вони різняться.

Текстовий потік – це послідовність рядків; кожна рядок має нуль або більше символів та закінчується символом '\n'. Операційне середовище може вимагати корекції текстового потоку (наприклад перекладу '\n' у символи повернення-каретки та переклад - рядки).  
Потік з'єднується з файлом або пристроєм за допомогою його відкриття, цей зв'язок розривається шляхом закриття потоку. Відкриття файлу повертає покажчик на об'єкт типу FILE, який містить усю інформацію, необхідну управління цим потоком.

Основною командою для обробки файлів є:   
FILE \*fopen(const char \*filename, const char \*mode)

Де:

*-fopen* відкриває файл із заданим ім'ям і повертає потік або NULL, якщо спроба відкриття

виявилася невдалою. Режим mode припускає такі значення:  
-*const char \*mode є додатковим параметром файлу, основними зеред них є:*

*"r"* - текстовий файл відкривається для читання (від read (англ.) - читати);

*"w"* - текстовий файл створюється для запису; старий вміст (якщо він був) викидається (від

write (англ.) - писати);

*"а"* - текстовий файл відкривається або створюється для запису в кінець файлу (від append (англ.) - додавати);

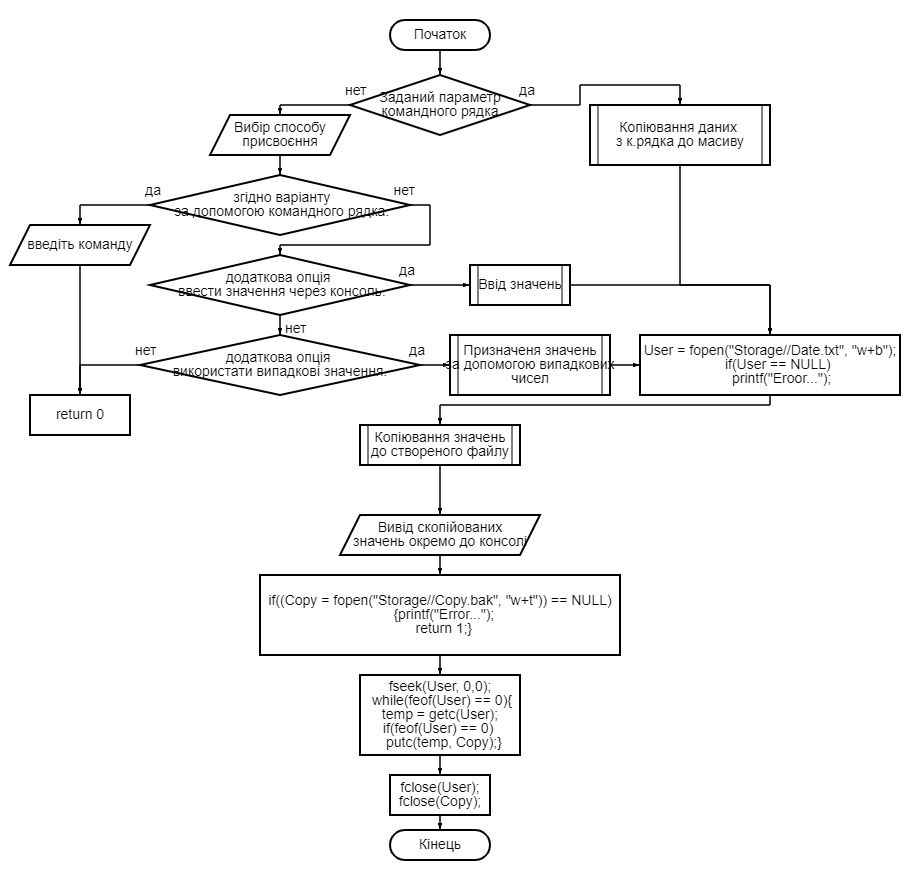
*"r+"* - текстовий файл відкривається для виправлення (тобто для читання та для запису);

*"w+"* – текстовий файл створюється для виправлення; старий вміст (якщо він був)

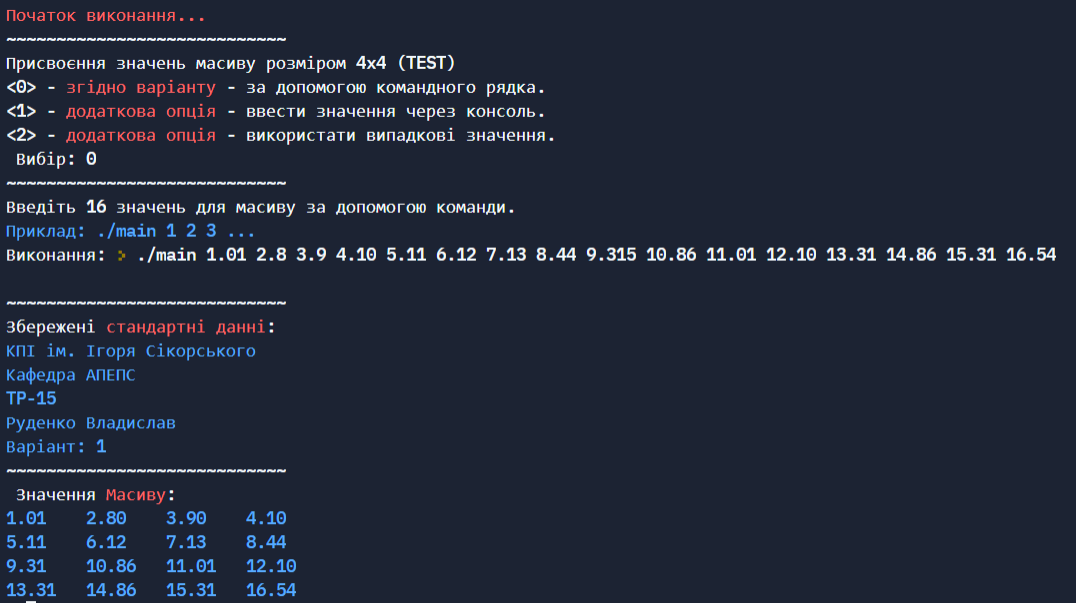
викидається;

*"а+"* - текстовий файл відкривається або створюється для виправлення вже існуючої інформації та додавання нової до кінця файлу.

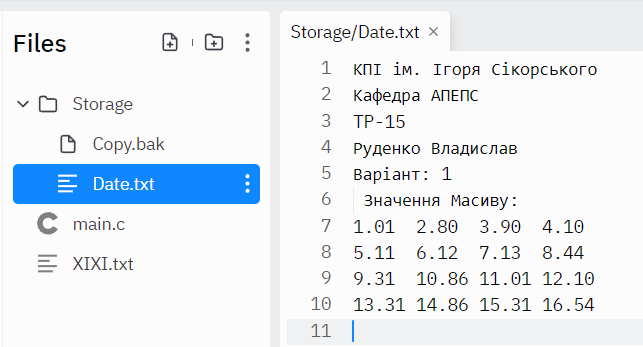
*Режим "виправлення" дозволяє читати і писати в той самий файл; під час переходів від операцій Читання до операцій запису та назад повинні здійснюватися звернення до fflush або до функції позиціонування файлу. Якщо вказівник режиму доповнити літерою b (наприклад, "rb" або "w+b"),це означатиме, що файл бінарний.*

***Блок схема алгоритму:  
***

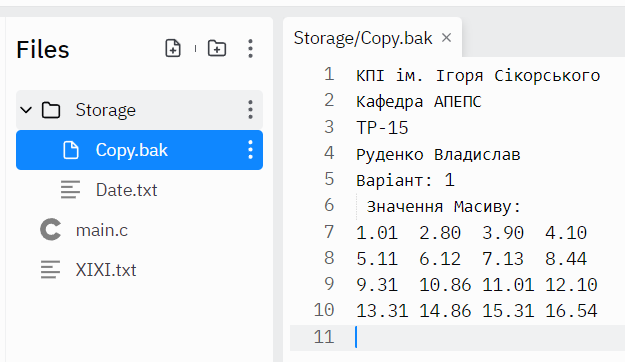
***Результати роботи:***



Результати роботи - 1 (Опрацювання в консолі)



Результати роботи - 2 (Скопійовані з консолі та початкові дані)



Результати роботи - 3 (Копіювання даних з одного файла до іншого)

***Хід виконаної роботи:***

Спочатку було створено перевірку на наявність додаткових параметрів командного рядка, після чого записано початкові дані для завдання, згодом було реалізовано декілька методів запису необхідних значень до масиву з типом “float”, це було зроблено для зручності перевірки ходу виконання програми в майбутньому. Після запису даних до масиву розміром 4х4 створюється (та перевіряється коректність створення) файлу для запису необхідних даних, після чого реалізується сам запис, спочатку початкові дані, а потім дані масиву.  
Як тільки копіювання даних завершено вказівник файлу переноситься на початок для подальшого копіювання даних з створеного файлу в інший, але з розширенням “bak”. Створюється новий файл (та перевірка коректного створення) файлу з розширенням “bak” після чого відбувається копіювання даних. Після завершення операції відбувається розрив зв’язку (закриття файлів).

***Посилання на програму в repl.it:***

***<https://replit.com/join/eweyxdweqt-hetik>***

***Висновки:***

Під час виконання лабораторної роботи №1 було ознайомлено функції для роботи з файлами. В якості індивідуального завдання було написано програмний код що реалізує створення двовимірного квадратного масиву розмірністю SIZE; виконання початкової ініціалізації всіх елементів масиву; запис у файл початкових даних і значень всіх елементів двовимірного квадратного масиву; створює копію цього файлу з новим розширенням \*.bak. Виведено скопійовані до файлу значення до консолі. Реалізовано декілька методів циклу в різних частинах коду. Також було опрацьовано виявлені в ході роботи помилки.

***Програмний код (у текстовому варіанті):****#include <stdlib.h>*

*#include <stdio.h>*

*#define RED "\x1b[31m"*

*#define BLUE "\x1b[34m"*

*#define YELLOW "\033[1;33m"*

*#define RESET "\x1b[0m"*

*#define size 4*

*#define WordSize 64*

*#define MassSize 1024*

*struct university*

*{*

*char FullName[WordSize];*

*char Department[WordSize];*

*char GroupName[WordSize];*

*char StudName[WordSize];*

*int variant[WordSize];*

*};*

*int main(int argc, char \*argv[])*

*{*

*////////////////////////////*

*FILE \*User, \*Copy, \*XIXI;*

*char temp;*

*int timer = 0, egg = 0;*

*srand(time(0));*

*////////////////////////////*

*struct university KPI;*

*/////////////////////////////////////////*

*strcpy(KPI.FullName,"КПІ ім. Ігоря Сікорського");*

*strcpy(KPI.Department, "Кафедра АПЕПС");*

*strcpy(KPI.GroupName, "TP-15");*

*strcpy(KPI.StudName, "Руденко Владислав");*

*strcpy(KPI.variant, "Варіант: 1");*

*/////////////////////////////////////////*

*float Values[size][size];*

*if(argc == 17)*

*{*

*printf("\n~~~~~~~~~~~~~~~~~~~~~~~~~~~~\n");*

*int temp = 1;*

*for(int i=0;i<size;i++)*

*for(int j=0;j<size;j++)*

*{*

*Values[i][j]=atof(argv[temp]);*

*temp++;*

*}*

*goto Metka;*

*}*

*/////////////////////////////////////////*

*int choose;*

*printf(RED"Початок виконання..."RESET"\n~~~~~~~~~~~~~~~~~~~~~~~~~~~~\n");*

*if (argc == 1)*

*{*

*start:*

*printf("Присвоєння значень масиву розміром %ix%i (TEST)\n<0> - "RED"згідно варіанту "RESET"- за допомогою командного рядка.\n<1> - "RED"додаткова опція"RESET" - ввести значення через консоль.\n<2> -"RED" додаткова опція "RESET"- використати випадкові значення. \n Вибір: ", size, size);*

*scanf("%i", &choose);*

*switch(choose)*

*{*

*case 0:*

*printf("~~~~~~~~~~~~~~~~~~~~~~~~~~~~\nВведіть 16 значень для масиву за допомогою команди.\n"BLUE"Приклад: ./main 1 2 3 ..."RESET"\nВиконання: ");*

*return 0;*

*case 1:*

*printf("~~~~~~~~~~~~~~~~~~~~~~~~~~~~\n");*

*for(int i=0;i<4;i++)*

*{*

*printf("Введіть "RED"%i"RESET" значення для рядка "BLUE"%i"RESET": ", size, i+1);*

*for(int j=0;j<4;j++)*

*scanf("%f", &Values[i][j]);*

*}*

*break;*

*case 2:*

*for(int i=0;i<size;i++)*

*for(int j=0;j<size;j++)*

*Values[i][j] = -20 + rand() %70;*

*break;*

*case 15:*

*egg++;*

*break;*

*}*

*printf("~~~~~~~~~~~~~~~~~~~~~~~~~~~~\n");*

*}*

*else*

*{*

*printf(RED"Error..."RESET"Ви допустили помилку, значень не 16.");*

*return 0;*

*}*

*if(egg==1)*

*{*

*printf(RED"\n\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n"RESET);*

*XIXI = fopen("XIXI.txt", "r");*

*fseek(XIXI, 0,0);*

*while(feof(XIXI) == 0)*

*{*

*temp = getc(XIXI);*

*if(feof(XIXI) == 0)*

*printf("%c",temp);*

*}*

*fclose(XIXI);*

*printf(RED"\n\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n\n"RESET);*

*egg--;*

*goto start;*

*}*

*Metka:*

*/////////////////////////////////////////*

*User = fopen("Storage//Date.txt", "w+b");*

*if(User == NULL)*

*printf(RED"Eroor..."RESET"Помилка створення файлу");*

*/////////////////////////////////////////////////////////*

*fprintf(User,"%s \n%s \n%s \n%s \n%s\n Значення Масиву:\n", KPI.FullName, KPI.Department, KPI.GroupName, KPI.StudName, KPI.variant );*

*for(int i=0;i<size;i++){*

*for(int j=0;j<size;j++)*

*fprintf(User,"%3.2f\t", Values[i][j]);*

*fprintf(User, "\n");*

*}*

*//////////////////////////////////////////////////////////*

*printf("Збережені "RED"стандартні данні"RESET":"BLUE" \n%s \n%s \n%s \n%s \n%s"RESET"\n~~~~~~~~~~~~~~~~~~~~~~~~~~~~\n Значення "RED"Масиву"RESET":\n", KPI.FullName, KPI.Department, KPI.GroupName, KPI.StudName, KPI.variant );*

*for(int i=0;i<size;i++){*

*for(int j=0;j<size;j++)*

*printf(BLUE"%3.2f\t"RESET, Values[i][j]);*

*printf("\n");*

*}*

*/////////////////////////////////////////////////////////*

*if((Copy = fopen("Storage//Copy.bak", "w+t")) == NULL)*

*{*

*printf(RED"\nУпс....Відкриття файлу не вдалося\n"RESET);*

*return 1;*

*}*

*//////////////////////////////////////////////////////////*

*fseek(User, 0,0);*

*//////////////////////////////////////////////////////////*

*while(feof(User) == 0)*

*{*

*temp = getc(User);*

*if(feof(User) == 0)*

*putc(temp, Copy);*

*}*

*//////////////////////////////////////////////////////////*

*fclose(User);*

*fclose(Copy);*

*//////////////////////////////////////////////////////////*

*}*