

Національний технічний університет України  
«Київський політехнічний інститут імені Ігоря Сікорського»  
Кафедра теоретичних основ радіотехніки

**ЗВІТ ДО ЛАБОРАТОРНОЇ РОБОТИ №5**

“Робота з файлами ”

з дисципліни: «Інформатика 1»

	Виконав:Луцкевич Володимир Андрійович	
--	--	--

	Група: РЕ-12
	Викладачі: доцент Катін П.Ю.
	Оцінка:
	Підпис:

Київ – 2021

**Мета:** отримати навички роботи з текстовими та двійковими (бінарними) файлами в мові C, форматним та неформатним вводом-виводом.

Код:

```
#define _CRT_SECURE_NO_WARNINGS
#include <conio.h>
#include <stdio.h>
#include <stdlib.h>
#include <process.h>

#define size 255

int main()
{
    //FILE* file = fopen("D:\Visual_Studio_2019\Lab_5_VL\Lab_5_VL\input3.txt", "a+");

    //fprintf(file, "test1");

    //fclose(file);

    //return 0;
//}

//char c;
float fp, fp2, fp3;
long l;
int i;
char line[size]; // char-масив, line-назва масиву
system("chcp 1251");
//     FILE* S1, * S2;
//     int x, y;
```

```

FILE* file = fopen("D:\\Visual_Studio_2019\\Lab_5_VL\\Lab_5_VL\\input3.txt", "a"); // вказівник типу даних файл ,
file- назва вказівника, fopen - функція повертає весь відкритий файл, 1 параметр функції - який файл або де він
знаходиться,
//2 параметр - метод відкриття файлу "r" читати "w" запис "a" запис без видалення інформації,
//"r+" - відкрити файл для читання та запису (файл повинен існувати);
//"w+" - відкрити порожній файл для читання та запису; якщо файл існує, його вміст
//"a+" - відкрити файл для читання та доповнення, якщо файл не існує, то він створюється
//звідси зчитуються дані

//fprintf(file, "1");

if ((file = fopen("input3.txt", "r+")) == NULL) //NULL-нульовий вказівник
{
    printf("error");
    //getchar();
    return 0;
}
else
{
    printf("file was opened");
}
for (i = 0; i < 5; i++)
{
    fgets(line, 255, file); // 1 парамет - масив з яким працюємо, 2 параметр - кількість символів які хочемо
зчитати, 3 параметр - з яким файлом працюємо
    //функція fgets() працює построково - перший виклик = 1 рядок, другий виклик = 2 рядок(можна виконати у
вигляді циклу)
    printf("\n\n%s", line);
}
//
//The prototype functions fseek :
//1
//
////int fseek(FILE * filestream, long int offset, int origin);
//SEEK_SET      Top of file
//SEEK_CUR      The current position of the file
//SEEK_END      End of file
// Встановити покажчик на початок файлу:
fseek(file, 0L, SEEK_SET); //0L - зміна позиції (у байт) відносно початку файла

// Read data back from file:
fscanf(file, "%s", line);
fscanf(file, "%ld", &l);
fscanf(file, "%f", &fp);
fscanf(file, "%f", &fp2);
fscanf(file, "%f", &fp3);
//fscanf(file, "%c", &c);
// Note: fscanf is deprecated; consider using fscanf_s instead

// Output data read:
printf("\n Line = %s\n", line);
printf("x1 = %ld\n", l);
printf("x2 = %f\n", fp);
printf("N = %f\n", fp2);
printf("d = %f\n", fp3);
//printf("N = %c\n", c);

```

```

///-----
//int f;
//
//fscanf(file, "%s%d", line, &f);
////&a вказуємо адресу куди записується значення
//
//
//int* b;
//b = &f;
//
////n1 = getw
//
////fprintf(file, "test2");
//printf("X1 = %d\n", *b);
// треба закривати файл коли робота з ним закінчена
fclose(file);

FILE* fileTxt = fopen("result.txt", "w");

//      fprintf(fileTxt, "\nGroup");

fprintf(fileTxt, "\n Group - %s\n", line);
fprintf(fileTxt, " X1 = %ld\n", l);
fprintf(fileTxt, " X2 = %f\n", fp);
fprintf(fileTxt, " N= %f\n", fp2);
fprintf(fileTxt, " d = %f\n", fp3);
fprintf(fileTxt, "You type:: %s %ld %s %f %s %f\n", "\n initial argument value x1 - ", l, "\n last argument value x2-",
fp, "\n value of the number of points N -", fp2);

fprintf(fileTxt, "-----\n");
fprintf(fileTxt, "    N | X |    \n"
"-----+-----+-----\n");
fprintf(fileTxt, "%s %f %s %f %s %ld %s ", " ", fp2, "    |", fp, "|", l, "|", "\n"
"-----+-----+-----\n");
fprintf(fileTxt, "        |        |        \n"
"-----+-----+-----\n");

fclose(fileTxt);

///-----

//3) Зчитування бінарного файлу
FILE* filebin = NULL;
int number;

filebin = fopen("result.bin", "wb");
if (filebin == NULL) {
    printf("Error opening file");
    //_getch();
    return 0;
}

```

```

fprintf(filebin, "\n Group - %s\n", line);
fprintf(filebin, " X1 = %ld\n", l);
fprintf(filebin, " X2 = %f\n", fp);
fprintf(filebin, " N= %f\n", fp2);
fprintf(filebin, " d = %f\n", fp3);
fprintf(filebin, "You type:: %s %ld %s %f %s %f\n", "\n initial argument value x1 - ", l, "\n last argument value x2-",
fp, "\n value of the number of points N -", fp2);

//fprintf(fileTxt, "F(x) = 2 * x1, F(x) = 2 * x2\n");

fprintf(filebin, "-----\n");
fprintf(filebin, "      N | X |      \n"
"-----+-----\n");
fprintf(filebin, "%s %f %s %f %s %ld %s", " ", fp2, "      |", fp, "|", l, "|", "\n"
"-----+-----\n");
fprintf(filebin, "      |      |      \n"
"-----+-----\n");

/*scanf("%d", &number);
fwrite(&number, sizeof(int), 1, filebin);*/

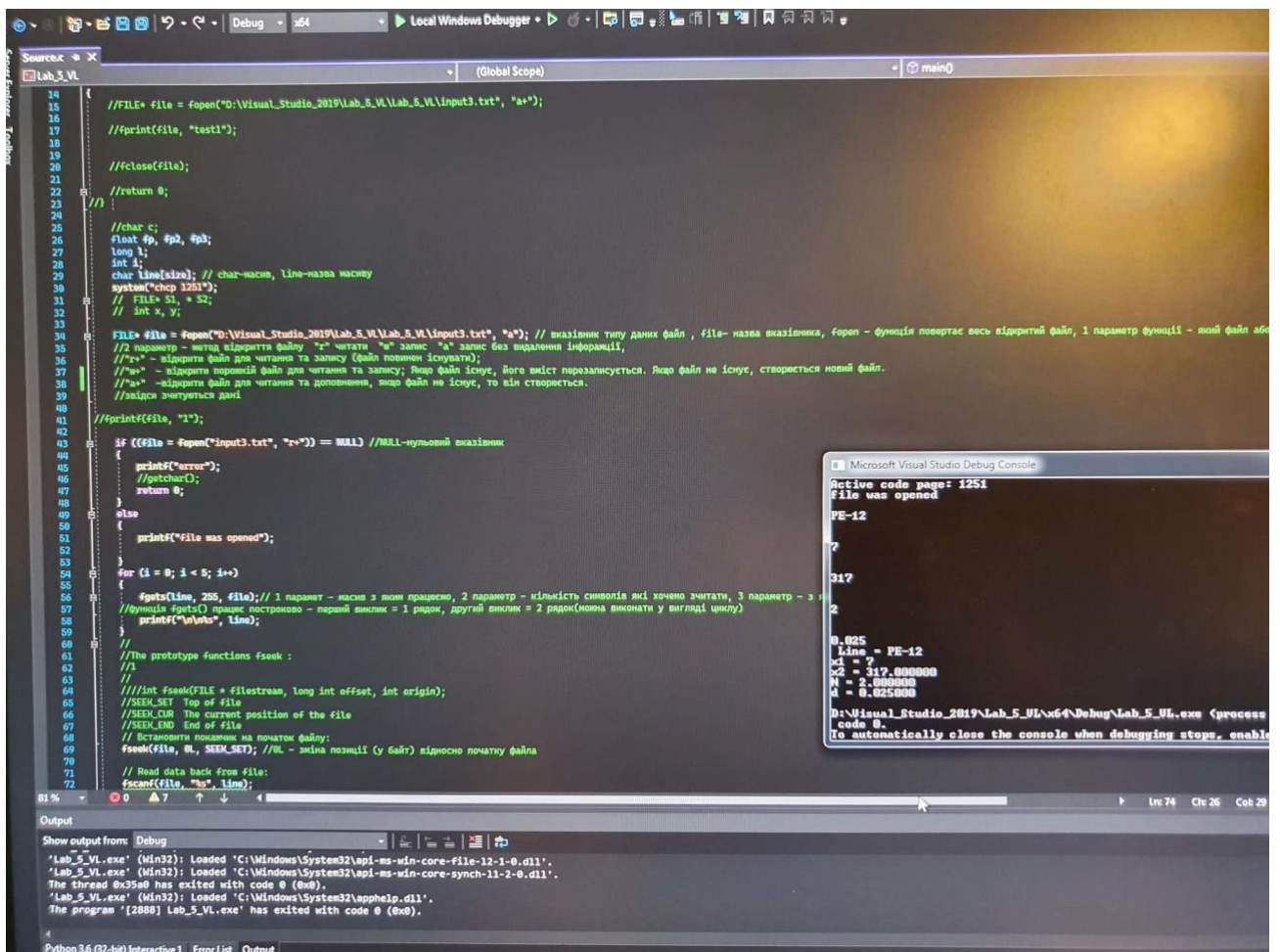
fclose(filebin);
//_getch();

return 0;
}

//result.bin size = 424 bytes
//result.txt size = 439 bytes

```

1) Вказуємо шлях файлу



- 2) Данні файлу до якого ми посилаємося виводяться на консоль

```
//функція fgets() працює построчно - перший виклик = 1 рядок, другий виклик = 2 рядок(можна виконати у вигляді циклу)
printf("\n\n%s", line);
}
//
//The prototype functions fseek :
//1
//
//int fseek(FILE * filestream, long int offset, int origin);
//SEEK_SET Top of file
//SEEK_CUR The current position of the file
//SEEK_END End of file
// Встановити покажчик на початок файлу:
fseek(file, 0L, SEEK_SET); //0L - зміна позиції (у байт) відносно початку файлу

// Read data back from file:
fscanf(file, "%s", line);
fscanf(file, "%ld", &l);
fscanf(file, "%f", &fp);
fscanf(file, "%f", &fp2);
fscanf(file, "%f", &fp3);
//fscanf(file, "%c", &c);
// Note: fscanf is deprecated; consider using fscanf_s instead

// Output data read:
printf("\n Line = %s\n", line);
printf("x1 = %ld\n", l);
printf("x2 = %f\n", fp);
printf("N = %f\n", fp2);
printf("d = %f\n", fp3);
//printf("N = %c\n", c);

//-----
//int f;
//
//fscanf(file, "%s", line, &f);
////&a вказуємо адресу куди записується значення
//
//
//int* b;
//b = &f;
//
//n1 = getw
//
//fprintf(file, "test2");
//printf("x1 = %d\n", *b);
// треба закривати файл коли робота з ним закінчена
fclose(file);
```

Microsoft Visual Studio Debug Console

Active code page: 1251  
file was opened

PE-12

7

317

2

0.025  
Line = PE-12  
x1 = 7  
x2 = 317.000000  
N = 2.000000  
d = 0.025000

D:\Visual\_Studio\_2019\Lab\_5\_UL\к64\Debug  
code 0.  
To automatically close the console when  
Debugging->Automatically close the conso  
Press any key to close this window . . .

- 3) Значення переносяться в файл "result.bin", "result.txt" у вигляді таблиці.







