

Лабораторна робота № 6. Функції

1 ВИМОГИ

1.1 Розробник

- Михневич Владислав Вікторович
- Студент 1-го курсу
- Групи КІТ-120а

1.2 Загальне завдання

Переробити програми, що були розроблені під час виконання лабораторних робіт з тем «Масиви» та «Цикли», таким чином, щоб використовувались функції для обчислення результату.

1.3 Додаткове завдання

Реалізувати функцію, що визначає, скільки серед заданої кількості чисел таких пар, у котрих перше число менше наступного, використовуючи функцію з варіативною кількістю аргументів

2 ОПИС ПРОГРАМИ

2.1 Функціональне призначення

В основу програми покладено, множення матриці саму на себе

3 РОЗРОБКА ПРОГРАМИ ТА ВАРІАНТИ ВИКОРИСТАННЯ

3.1 Структура

Створимо в папці свого репозиторію папку lab_07, в яку будемо додавати нові файли (див. рис.1).

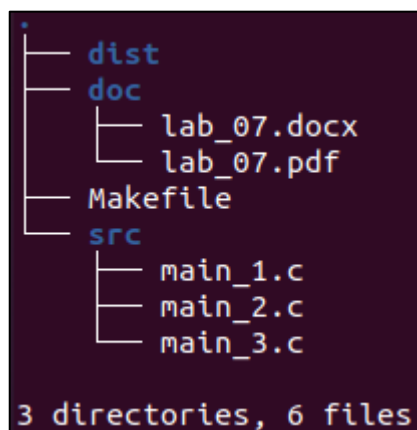


Рисунок 1 – Структура проекту

3.2 Початкові дані, типи даних

В нас є константа SIZE, її розмір 3. Дано дві функції int main - головна та void multiplication - яка буде використовуватися для обчислення результату, задали змінні matrix – в яку задаємо вхідні дані за допомогою генератора псевдовипадкових чисел rand () та result - в яку буде виводитися результат дій над матрицею (див. рис. 2).

```
#include <stdio.h>
#include <time.h>
#include <stdlib.h>

#define SIZE 3
void multiplication();

int main() {
    multiplication ();
    return 0;
}

void multiplication () {
    int matrix [SIZE][SIZE];
    int results [SIZE][SIZE]={0};

    srand(time(NULL));
    for (int i = 0; i < SIZE; i++) {
        for (int j = 0; j < SIZE; j++) {
            matrix [i][j] = (rand() % 15);
        }
    }

    for (int i = 0; i < SIZE; i++) {
        for (int j = 0; j < SIZE; j++) {
            for (int k = 0; k < SIZE; k++) {
                results [i][j] += matrix [i][k] * matrix [k][j];
            }
        }
    }
    return;
}
```

Рисунок 2 – Код програми

3.4 Компіляція та перевірка в debug

Зробимо компіляцію за допомогою команди `gcc -g main_2.c -o main_2.bin`, якщо наша програма написана вірно, то можемо переходити до `nemiver`, побачити значення змінних та процес виконання програми (див. рис. 3).

ID по	Кадр	Функция	Переменная	Значение	Тип
1	0	multiplication	Локальные переменные		
	1	main	matrix	[3]	int [3][3]
			0	[3]	int [3]
			0	1	int
			1	5	int
			2	13	int
			1	[3]	int [3]
			0	7	int
			1	0	int
			2	9	int
			2	[3]	int [3]
			0	6	int
			1	6	int
			2	6	int
			results	[3]	int [3][3]
			0	[3]	int [3]
			0	114	int
			1	83	int
			2	136	int
			1	[3]	int [3]
			0	61	int
			1	89	int
			2	145	int
			2	[3]	int [3]
			0	84	int
			1	66	int
			2	168	int
			Параметры функции		
Терминал цели Контекст Точки останова Регистры Память Монитор выражений					

Рисунок 3 – Робота в `nemiver`

Висновки

На цій лабораторній роботі ми попрацювали з функціями, а саме дізналися як представляються функції в мові програмування C. Використовували функцію з варіативною кількістю аргументів, навчилися передавати параметри функції, та повертати значення змінних через `return` та попрацювали з функцією `rand()`.