**Лабораторна робота №10. Документація проекту**

**1. Вимоги**

**1.1 Розробник**

Макаренко Владислав Олександрович;

студент групи КІТ-120а;

16-груд-2020

**1.2 Загальне завдання**

Розробити повноцінний звіт для лабораторної роботи “Функції”

**1.3 Індивідуальне завдання**

За допомогою функцій отримати корінь заданого числа.

**2. Опис програми**

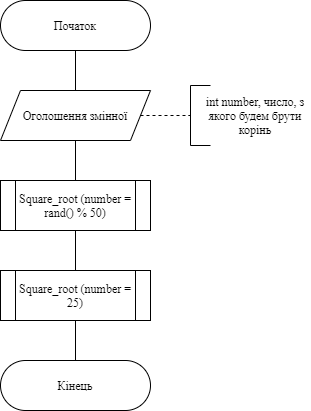
**2.1 Функціональне призначення**

Програма добуває квадратний корінь з числа за допомогою функції Square\_root(). Результат зберігається у змінній result. Демонстрація результату передбачає покрокове виконання програми.

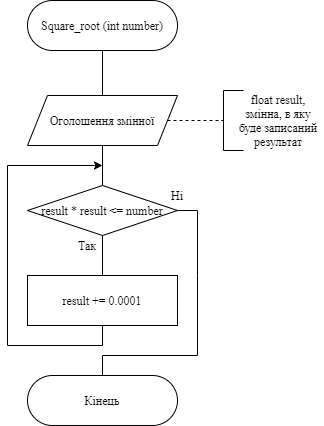
**2.2 Опис логічної структури програми**

Для визначення квадратного кореня з чилса викликаємо функцію Square\_root(), яка приймає параметр: число number з якого будет добуватись квадратний корінь. Функція збільшує значення параметру reult на 0.0001 поки значення result \* result не буде дорівнювати заданому number.

*Головна функція* main(). Приймає задане нами число. Задає випадкове число від 1 до 50. Двічі викликає функцію Square\_root. *Схема алгоритму функції* подана на рис. 1.

[](https://github.com/Vlad-Makarenko/Programing-repo/blob/main/lab10/doc/assets/main%20%28%29.png?raw=true)  
Рисунок 1 — Схема алгоритму функції main

*Функція, що добуває корінь з числа* Square\_root. Добуває квадратний корінь з числа. Параметри: number – число, з якого потрібно добути корінь; result – добутий корінь з заданого числа. Функція повертає result. *Схема алгоритму функції* подана на рис. 2.

[](https://github.com/Vlad-Makarenko/Programing-repo/blob/main/lab10/doc/assets/Square_root%28%29.png?raw=true)  
Рисунок 2 — Схема алгоритму функції Square\_rot

**Структура проекту**

├──lab10

├── Doxyfile

├── Makefile

├── doc

├── lab10.md

├── lab10.docx

├── lab10.pdf

└── assets

├── Square\_root().png

├── main ().png

├── main.c.png

├── result1.png

└── result2.png

└── src

└── main.c

**2.3 Важливі фрагменти програми**

**Підключення заголовочного stdlib.h та time.h для генерації випадкових чисел.**

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

**Виклик функції для випадкового числа**

result1 = Square\_root (number = rand() % 50);

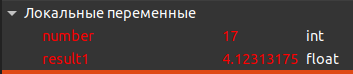
**Добування кореню за допомогою цикла**

while (result \* result <= number) {  
 result += 0.0001;  
}

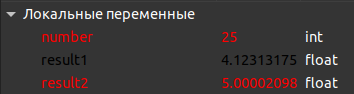
**3. Варіанти використання**

Для демонстрації результатів використовується покрокове виконання програми та інші засоби налагодження відлагодника nemiver. Нижче наводиться послідовність дій запуску програми у режимі відлагодження.

*Крок 1* (рис. 3). Виконуємо функцію для випадково згенерованого числа

[](https://github.com/Vlad-Makarenko/Programing-repo/blob/main/lab10/doc/assets/result1.png?raw=true)  
Рисунок 3 — Результат виконання функції для першого кроку

*Крок 2* (див. рис. 4). Виконуємо функцію для заданого нами числа

[](https://github.com/Vlad-Makarenko/Programing-repo/blob/main/lab10/doc/assets/result2.png?raw=true)  
Рисунок 4 — Результат виконання функції для другого кроку

**Висновки**

Ми навчились документувати проект за допомогою Markdown та в doc форматі, згідно ДСТУ.