

## Лабораторна робота №8.2

### Вступ до документації проекту

#### Розробник

Автор: Ветріченко Дарина Ігорівна

Група: КН-924з

Дата виконання: 06 грудня 2024 року

#### Загальне завдання

Розробити повноцінний звіт для лабораторної роботи №8 "Функції" у форматах Markdown та DOCX згідно з ДСТУ. Реалізувати програму, яка виконує перевірку числа на простоту та підрахунок слів у тексті, з додаванням коментарів для Doxygen. Згенерувати документацію за допомогою Doxygen і додати її у звіт.

#### Опис програми

##### Функціональне призначення

Програма виконує дві основні функції:

1. Перевіряє, чи є задане число простим.
2. Підраховує кількість слів у тексті.

##### Опис логічної структури

1. Функція isPrime:

- Приймає ціле число як параметр.
- Повертає 1, якщо число просте, або 0, якщо не просте.

2. Функція countWords:

- Приймає текст у вигляді рядка символів.
- Аналізує кожен символ тексту, визначаючи слова, і повертає кількість слів.

3. Основна функція:

- Приймає аргументи командного рядка:
  - Перше число — перевіряється на простоту.
  - Другий текст — використовується для підрахунку слів.
- Генерує випадкові дані для демонстрації роботи функцій.

#### Структура проекту

```
.
├── lab08
│   ├── README.md
│   ├── Doxyfile
│   └── Makefile
```

```
├── src
│   └── main.c
└── doc
    ├── lab08.md
    ├── lab08.docx
    ├── lab08.pdf
    └── assets/
```

## Важливі фрагменти програми

### Функція *isPrime*:

```
int isPrime(int number) {
    if (number < 2) return 0; // Число не є простим
    for (int i = 2; i * i <= number; i++) {
        if (number % i == 0) return 0; // Знайдено дільник
    }
    return 1; // Число просте
}
```

### Функція *countWords*:

```
int countWords(const char *text) {
    int wordCount = 0;
    int inWord = 0;
    while (*text) {
        if (isspace(*text)) {
            inWord = 0; // Вийшли зі слова
        } else if (!inWord) {
            inWord = 1; // Увійшли в слово
            wordCount++;
        }
        text++;
    }
    return wordCount;
}
```

## Варіанти використання

### 1. Перевірка числа на простоту:

./dist/main 17

Результат:

Перевірка числа 17: Просте

### 2. Підрахунок слів у тексті:

```
./dist/main "Привіт, це демонстраційний текст."
```

Результат:

Текст: Привіт, це демонстраційний текст.

Кількість слів: 4

3. Генерація документації:

```
make doxygen
```

Документація зберігається в dist/html.

### **Висновки**

1. У ході виконання лабораторної роботи було створено програму, яка виконує задачі перевірки числа на простоту та підрахунку слів у тексті.
2. Використано систему Doxygen для документування та автоматичної генерації документації.
3. Звіт виконано у форматах Markdown та DOCX згідно з вимогами.