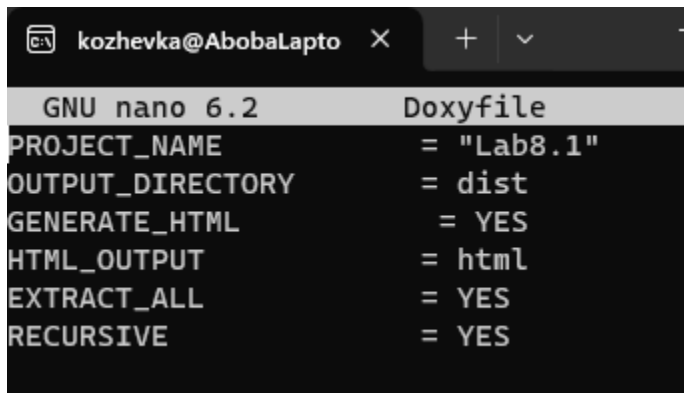
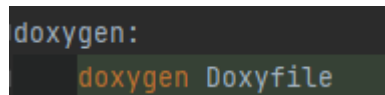


1. Команда “sudo apt-get install doxygen mscgen graphviz” встановлює утиліти. Дозволяю введенням пароллю - завантажую - підтверджую встановлення “Y”
2. Редактором nano створюю Doxyfile у директорії проекту.



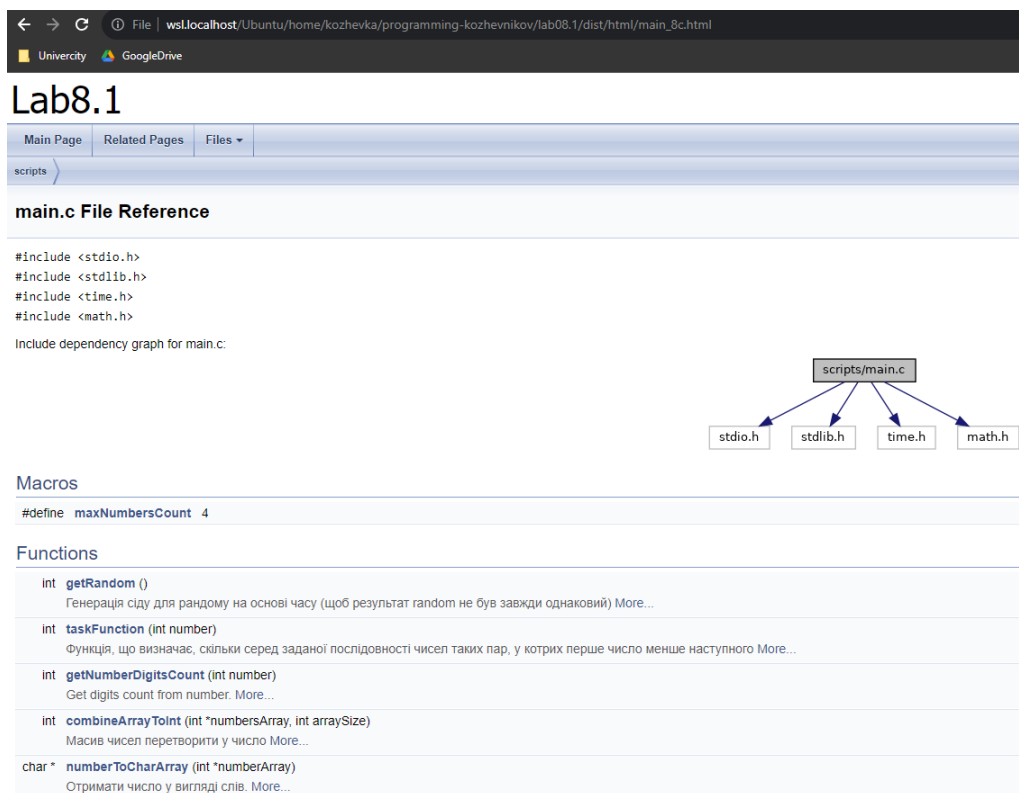
```
kozhnevka@AbobaLapto x + v
GNU nano 6.2 Doxyfile
PROJECT_NAME           = "Lab8.1"
OUTPUT_DIRECTORY       = dist
GENERATE_HTML           = YES
HTML_OUTPUT            = html
EXTRACT_ALL            = YES
RECURSIVE               = YES
```

3. Таким же чином створив README.md файл с довільним контентом.
4. Оновив коментарі у коді щоб відповідали умовам doxyfile



```
doxygen:
doxygen Doxyfile
```

5. У makefile додав `doxygen Doxyfile` і додав операцію doxygen до команди `all`. `all: clean prep compile doxygen check`
6. Збілдив проект і перевірів документацію на прикладі main.c:



Lab8.1

Main Page | Related Pages | Files ▾

scripts ▸

main.c File Reference

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

Include dependency graph for main.c:

```
graph TD
    scripts/main.c --> stdio.h
    scripts/main.c --> stdlib.h
    scripts/main.c --> time.h
    scripts/main.c --> math.h
```

Macros

```
#define maxNumbersCount 4
```

Functions

| | | |
|--------|------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------|
| int | getRandom () | Генерація сіду для рандому на основі часу (щоб результат random не був завжди однаковий) More... |
| int | taskFunction (int number) | Функція, що визначає, скільки серед заданої послідовності чисел таких пар, у котрих перше число менше наступного More... |
| int | getNumberDigitsCount (int number) | Get digits count from number. More... |
| int | combineArrayToInt (int *numbersArray, int arraySize) | Масив чисел перетворити у число More... |
| char * | numberToCharArray (int *numberArray) | Отримати число у вигляді слів. More... |

7. Додав lab08.1/dist/ до .gitignore файлу

Відповіді на питання :

Що таке система Doxygen?

Doxygen - це система автоматизованої генерації документації з коментарів у програмному коді. Вона дозволяє отримати структуровану та зрозумілу документацію для програмного проекту на основі коментарів, вбудованих у вихідний код.

Які результуючі формати документації підтримує Doxygen?

Doxygen підтримує різні формати вихідної документації, включаючи HTML, LaTeX, RTF, XML, Man pages (Unix), а також можливість генерації графічних зображень залежностей за допомогою Graphviz.

Як у консольному режимі згенерувати документацію до проекту? Що для цього потрібно?

Для генерації документації в консольному режимі за допомогою Doxygen, вам потрібно викликати команду `doxygen` та передати шлях до конфігураційного файлу `Doxyfile` або вказати параметри безпосередньо в командному рядку.

Призначення утиліти dot.

Утиліта `dot` є частиною пакету Graphviz і використовується Doxygen для генерації графічних зображень залежностей, таких як діаграми класів та графи викликів функцій.

Призначення пакету mscgen.

Пакет `mscgen` використовується Doxygen для генерації графічних представлень діаграм послідовностей (Message Sequence Charts).

Як у коді визначити, що поточний коментар потрапить до документації?

Для того, щоб коментар потрапив до документації, використовуйте спеціальні теги, такі як `/** ... */` для C/C++ або `/*! ... */` для C++.

Які команди необхідні для документування файлу взагалі?

Для того, щоб коментарі потрапили до документації, використовуйте теги, наприклад, `/** ... */` у C/C++. Для документування функцій та змінних, вставляйте коментарі перед відповідними елементами коду.

Які механізми існують для документування загальної інформації про проект?

Загальна інформація про проект може бути задокументована в файлі README.md, який потім може бути включений до головної сторінки doxygen документації. Також, Doxygen дозволяє вставляти загальні описи проекту у конфігураційний файл Doxyfile.