

НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ  
«ХАРКІВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ»

Кафедра «Обчислювальної техніки та програмування»

Звіт з лабораторних робіт №8, 9, 10

Тема: «Вступ до блок-схем, документації коду, документації проекту»

Виконав:

ст. гр. КІТ-120в Львов Артем

Перевірив:

Бульба С.С.

Харків 2020

## Лабораторні роботи №8, 9, 10. Вступ до блок-схем, документації коду, документації проекту

### Розробник:

Львов Артем Сергійович

Студент групи КІТ-120В

23.12.2020

### Загальне завдання:

Для функцій, розроблених в лабораторних роботах 5, 6 та 7, побудувати блок-схеми алгоритмів, додати можливість генерації Doxygen документації, зробити Markdown звіти

### Виконання роботи:

#### 1. Створення репозиторію:

Створюю новий репозиторій, клоную в нього репозиторій однієї з попередніх лабораторних робіт, та клоную у систему командою `git clone`

#### 2. Коментування коду для подальшої коректної побудови Doxygen документації:

Пишу код програми та коментарі у файлі `Lab_08_09_10/src/main.c`

```
1 /**
2  * @mainpage
3  * @ Main Task
4  * 1. Create function to define that given number is prime or not.
5  *
6  * 2. Create function to change defined characters in string to make it a nickname.
7  *
8  * 3. Create function to define how many number pairs which have the second number bigger than the first are in given number
9  * sequence.
10 *
11 * @author Lvov A.
12 *
13 * @date 23-dec-2020
14 *
15 * @version 1.0
16 */
17
18 /**
19 * @file main.c
20 * @brief File that demonstrates how functions @function primenum , @function nickname and @function pairs work.
21 *
22 * @author Lvov A.
23 *
24 * @date 23-dec-2020
25 *
26 * @version 1.0
27 */
```

Рисунок 2.1 – Коментарі стосовно загальної структури проекту та файлу `main.c`

```

35 /**
36  * Main function
37  *
38  * Sequencing:
39  * - check that number is prime or not by calling function @function primenum
40  * - defining string and changing it to nickname by calling function @function nickname
41  * - calculate how many number pairs which have the second number bigger than the first are in given number sequence
42  * - @return successful returning of program with code (0)
43  */
44
45 int main()
46 {
47     int test_1 = primenum(rand() % 100);           /*test of "primenum()" function*/
48     char test_2[] = {"aois AOIS"};
49     nickname(test_2);                             /*test of "nickname()" function*/
50     int test_3 = pairs (4, 2 ,4 ,6 ,8);           /*test of "pairs()" function*/
51     return 0;
52 }

```

Рисунок 2.2 – об’явлення функції *int main()* з коментарями

```

54 /**
55  * @brief Function that defines number is prime or not
56  * @param One integer value
57  * @return 1 or 0
58  */
59
60 int primenum (int src)
61 {
62     int res = 0;
63     for (int i = 1; i < src; i++){                 /*defining variable for GF*/
64         if (src % i == 0){                         /*cycle to calculate the GF*/
65             res = i;
66         }
67     }
68     if (res == 1){
69         return 1;                                 /*returning result if the number is prime*/
70     }
71     else{
72         return 0;                                 /*returning result if the number is not prime*/
73     }
74 }

```

Рисунок 2.3 – об’явлення функції *int primenum()* з коментарями

```

76 /**
77  * @brief Function that changes defined symbols in string and turns it into a nickname
78  * @param String of symbol array
79  * @return Logical reply to console
80  */
81
82 void nickname (char src[])
83 {
84     for (int i = 0; i <= strlen(src); i++){        /*cycle for changing defined characters*/
85         if (src[i] == 'a' || src[i] == 'A'){
86             src[i] = 'e';
87         }
88         else if (src[i] == 'o' || src[i] == 'O'){
89             src[i] = '0';
90         }
91         else if (src[i] == 'i' || src[i] == 'I'){
92             src[i] = '1';
93         }
94         else if (src[i] == 's' || src[i] == 'S'){
95             src[i] = '5';
96         }
97     }
98 }

```

Рисунок 2.4 – об’явлення функції *void nickname()* з коментарями

```

100 /**
101  * @brief Function that defines how many number pairs which have the second number bigger than the first are in given number
102  * sequence
103  * @param Variative count of integer values (at least 2)
104  * @return Integer value
105  */
106
107 int pairs (const int cnt, ...)
108 {
109     int getres [cnt];                             /*defining array for given arguments*/
110     int result = 0;                                /*defining variable for resulting*/
111     va_list list;
112     va_start (list, cnt);
113     for (int i = 0; i <= cnt; i++){                 /*cycle to write arguments into the array*/
114         getres [i] = va_arg (list, int);
115     }
116     va_end (list);
117     for (int i = 1; i <= cnt; i++){                 /*cycle to count pairs*/
118         if (getres [i] < getres [i + 1]){
119             result ++;
120         }
121     }
122     if (getres [0] > getres [1]){                  /*check to fix result if the first arg is bigger than count*/
123         result = result - 1;
124     }
125     return result;                                 /*returning result*/
126 }

```

Рисунок 2.5 – об’явлення функції *int pairs()* з коментарями

## 2.1. Компіляція проекту:

Компілюю проект за допомогою команди `make clean prep compile`:

```
071@071:~/Lvov_Artem/Lab_08_09_10$ make clean prep compile
rm -rf dist
mkdir dist
gcc -std=gnu11 -g -Wall -Wextra -Wformat-security -Wfloat-equal -Wshadow -Wconversion -Wlogical-not-parentheses -Wnull-dereference src/main.c -o ./dist/main.bin
src/main.c: In function 'main':
src/main.c:47:24: warning: implicit declaration of function 'rand' [-Wimplicit-function-declaration]
   47 |     int test_1 = primum(rand() % 100); /*test of "primum()" function*/
      |                        ^
src/main.c:50:6: warning: unused variable 'test_3' [-Wunused-variable]
   50 |     int test_3 = pairs(4, 2, 4, 6, 8); /*test of "pairs()" function*/
      |
src/main.c:47:6: warning: unused variable 'test_1' [-Wunused-variable]
   47 |     int test_1 = primum(rand() % 100); /*test of "primum()" function*/
      |
src/main.c: In function 'nickname':
src/main.c:84:29: warning: 'sizeof' on array function parameter 'src' will return size of 'char *' [-Wsizeof-array-argument]
   84 |     for (int i = 0; i <= sizeof(src); i++){ /*cycle for changing defined characters*/
      |                             ^
src/main.c:82:21: note: declared here
   82 |     void nickname(char src[])
      |
src/main.c:84:20: warning: comparison of integer expressions of different signedness: 'int' and 'long unsigned int' [-Wsign-compare]
   84 |     for (int i = 0; i <= sizeof(src); i++){ /*cycle for changing defined characters*/
      |
071@071:~/Lvov_Artem/Lab_08_09_10$
```

Рисунок 2.1.1 – компіляція проекту

## 3. Завантаження проекту на GitHub:

```
071@071:~/Lvov_Artem/Lab_08_09_10$ git add .
071@071:~/Lvov_Artem/Lab_08_09_10$ git commit -m"all"
[master 7df0f54] all
15 files changed, 421 insertions(+)
create mode 100644 Lab_08_09_10/Doxyfile
create mode 100644 Lab_08_09_10/Makefile
create mode 100644 Lab_08_09_10/doc/assets/Lab_05.png
create mode 100644 Lab_08_09_10/doc/assets/Lab_06.png
create mode 100644 Lab_08_09_10/doc/assets/Lab_07.png
create mode 100644 Lab_08_09_10/md/Lab_05_m.md
create mode 100644 Lab_08_09_10/md/Lab_06_m.md
create mode 100644 Lab_08_09_10/md/Lab_07_m.md
create mode 100644 Lab_08_09_10/md/flowcharts/Lab_05_m.png
create mode 100644 Lab_08_09_10/md/flowcharts/Lab_06_m.png
create mode 100644 Lab_08_09_10/md/flowcharts/Lab_07_m.png
create mode 100644 Lab_08_09_10/md/output_results/Lab_05_m.png
create mode 100644 Lab_08_09_10/md/output_results/Lab_06_m.png
create mode 100644 Lab_08_09_10/md/output_results/Lab_07_m.png
create mode 100644 Lab_08_09_10/src/main.c
071@071:~/Lvov_Artem/Lab_08_09_10$ git push
Username for 'https://github.com': sudo-071
Password for 'https://sudo-071@github.com':
Enumerating objects: 25, done.
Counting objects: 100% (25/25), done.
Compressing objects: 100% (22/22), done.
Writing objects: 100% (24/24), 344.92 KiB | 14.37 MiB/s, done.
Total 24 (delta 3), reused 0 (delta 0)
remote: Resolving deltas: 100% (3/3), completed with 1 local object.
To https://github.com/sudo-071/Lvov_Artem.git
   1481585..7df0f54 master -> master
071@071:~/Lvov_Artem/Lab_08_09_10$
```

Рисунок 3 – Виконання команд `git add`, `git commit`, `git push`

#### 4. Наведу графічні блок-схеми роботи даної програми:

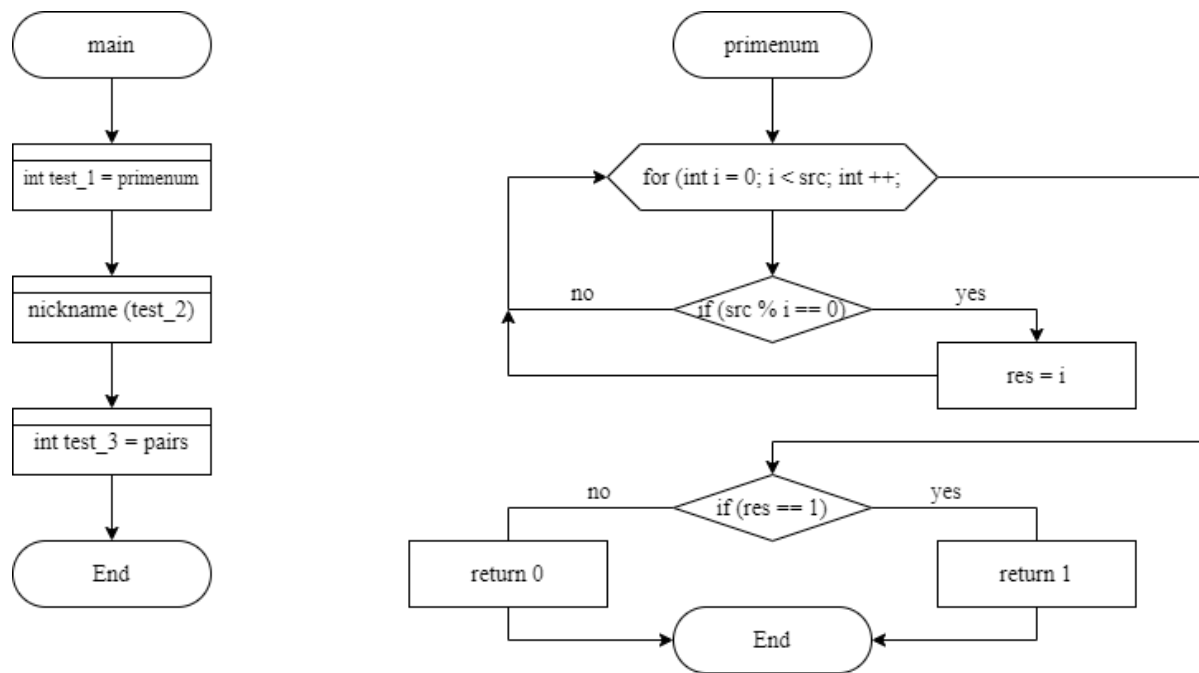


Рисунок 4.1 – блок-схема функцій *main()* та *primenum()*

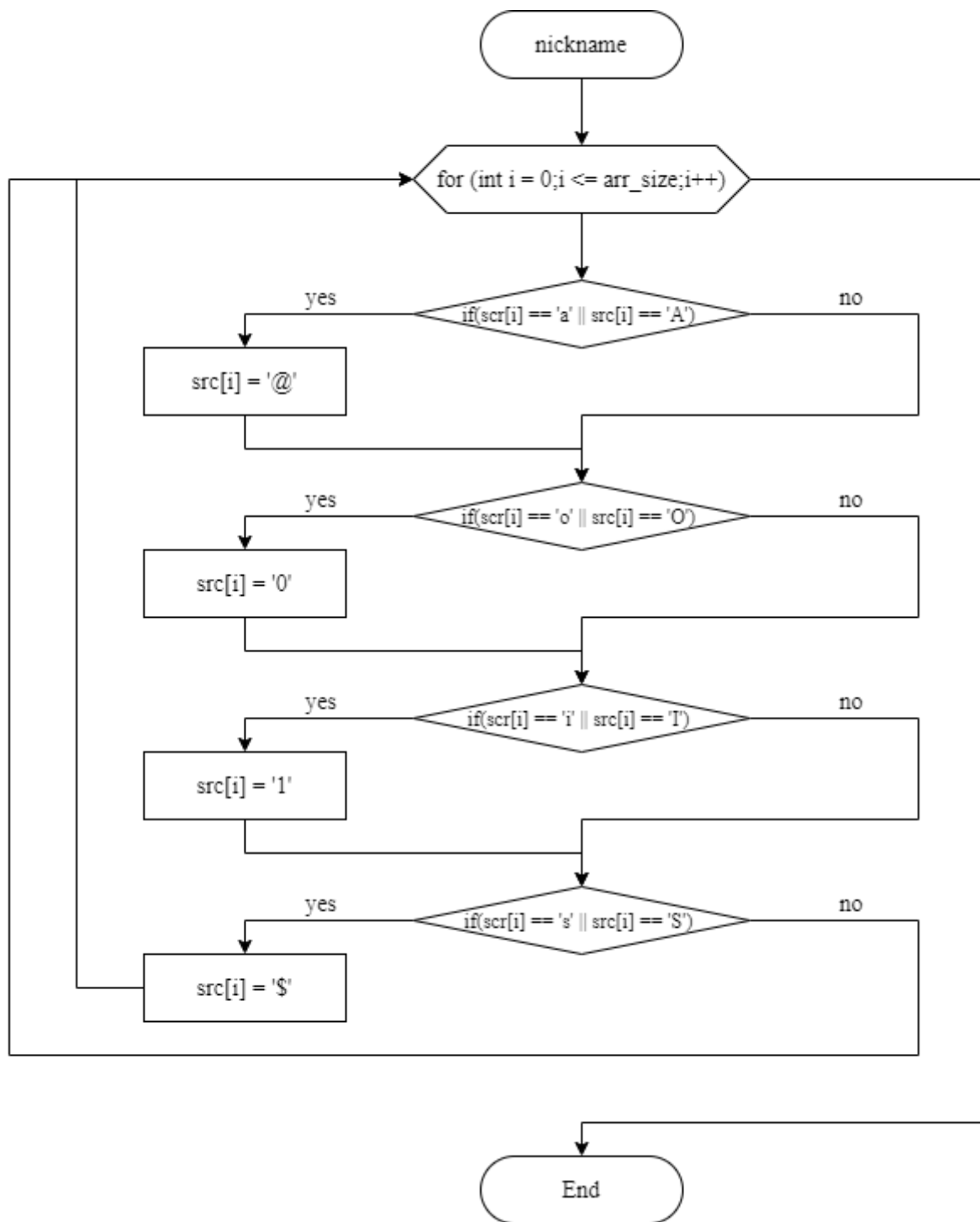


Рисунок 4.2 – блок-схема функції *nickname()*

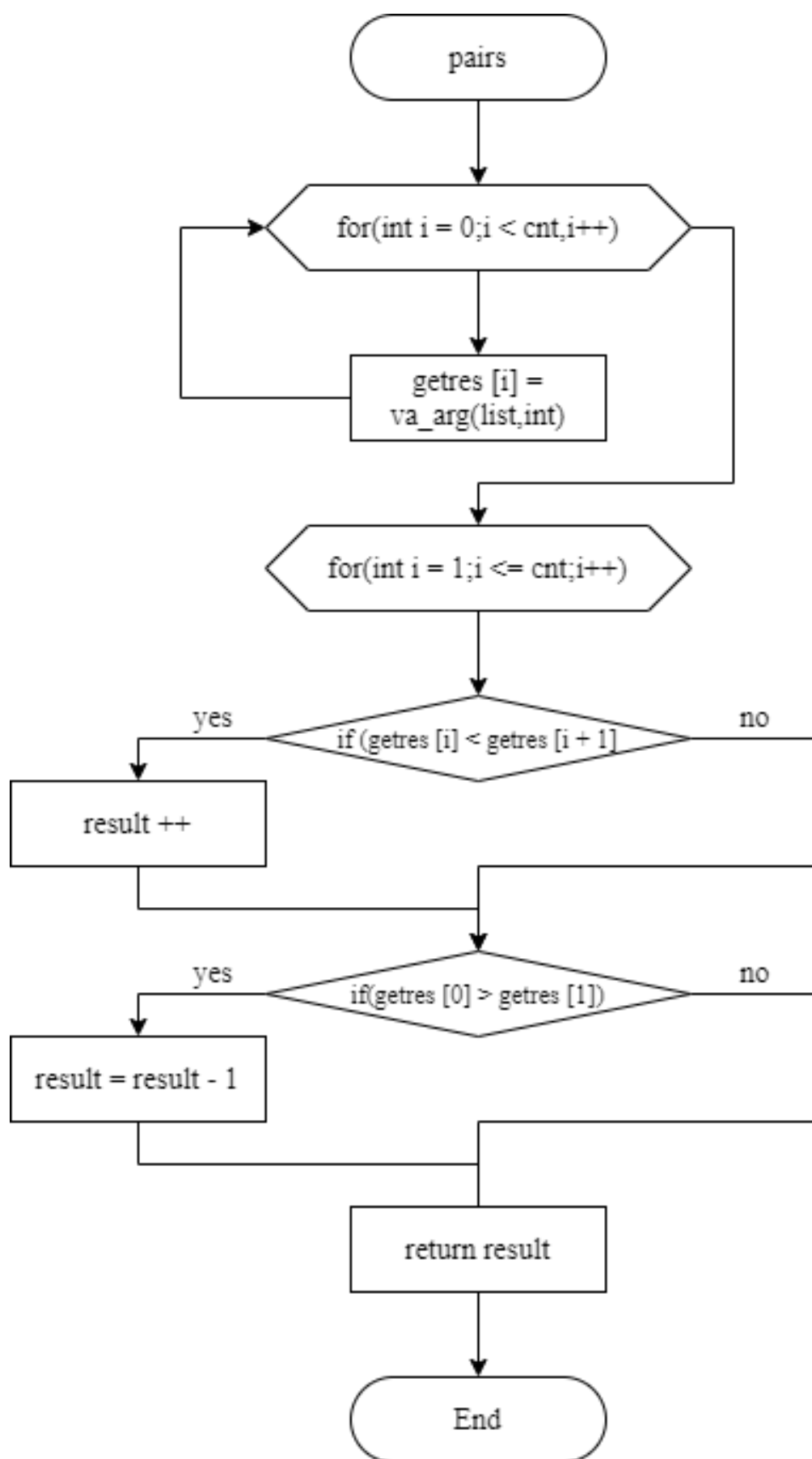


Рисунок 4.3 – блок-схема функції *pairs()*

**Висновки:** Під час виконання цих лабораторних робіт, я навчився будувати блок-схеми алгоритмів функцій, додавати до коду коментарі для подальшої генерації Doxygen документації, та робити звіти в форматі Markdown.