

Міністрество освіти і науки України
Національний технічний університет
«Харківський політехнічний інститут»
Кафедра «Обчислювальна техніка та програмування»

ЗВІТ

Про виконання лабораторної роботи №5
« Циклічні конструкції »

Виконавець: Мокрослоєв А.В. (КІТ-120є)
Перевірив викладач: Бульба С. С.

Харків 2020

Лабораторна робота №5. Циклічні конструкції

1. Вимоги

1.1 Розробник

Студент групи КІТ-120є Мокрослоєв Антон Вікторович

1.2 Загальне завдання

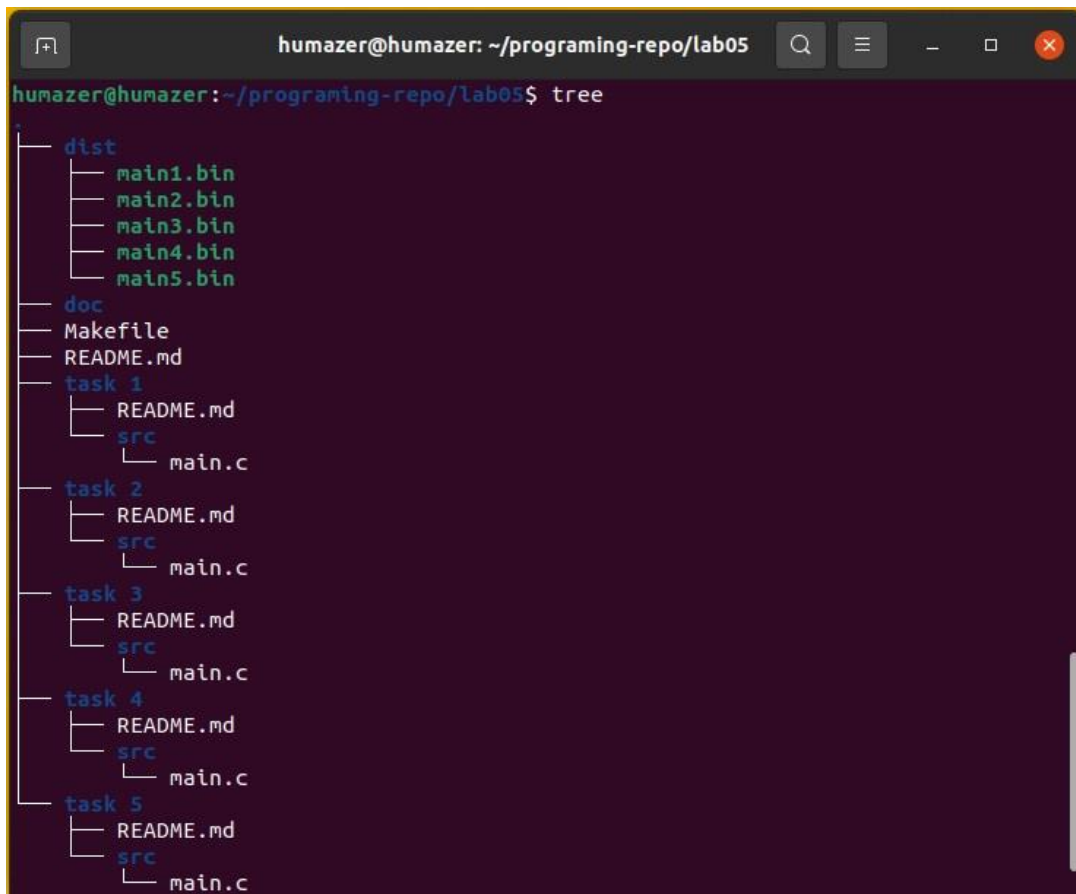
Виконати всі завдання з категорії: "відмінно". Реалізувати програму відповідно до індивідуального завдання за допомогою трьох типів циклів: for, while-do, do-while.

Індивідуальне завдання:

№1 Визначити найбільший спільний дільник для двох заданих чисел.

2. Опис програми

2.1 Було створено структуру проекту (Рис. 1), створено папку *lab05*, файл *Makefile* для компіляції та папку *src* для вихідного коду.

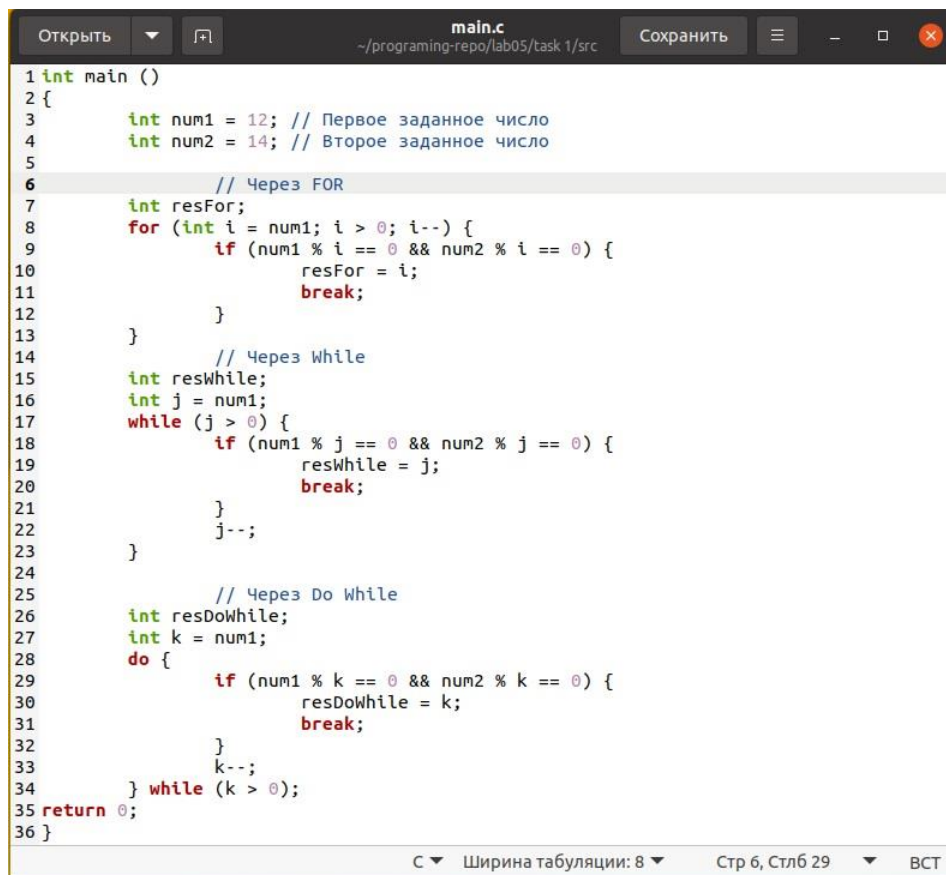


```
humazer@humazer: ~/programming-repo/lab05
humazer@humazer:~/programming-repo/lab05$ tree
.
├── dist
│   ├── main1.bin
│   ├── main2.bin
│   ├── main3.bin
│   ├── main4.bin
│   └── main5.bin
├── doc
│   ├── Makefile
│   └── README.md
├── task 1
│   ├── README.md
│   └── src
│       └── main.c
├── task 2
│   ├── README.md
│   └── src
│       └── main.c
├── task 3
│   ├── README.md
│   └── src
│       └── main.c
├── task 4
│   ├── README.md
│   └── src
│       └── main.c
└── task 5
    ├── README.md
    └── src
        └── main.c
```

Рисунок 1 – Структура проекту

2.2 Виконано задачі 1-5, код збережено у файлах *task1.c* – *task5.c*.

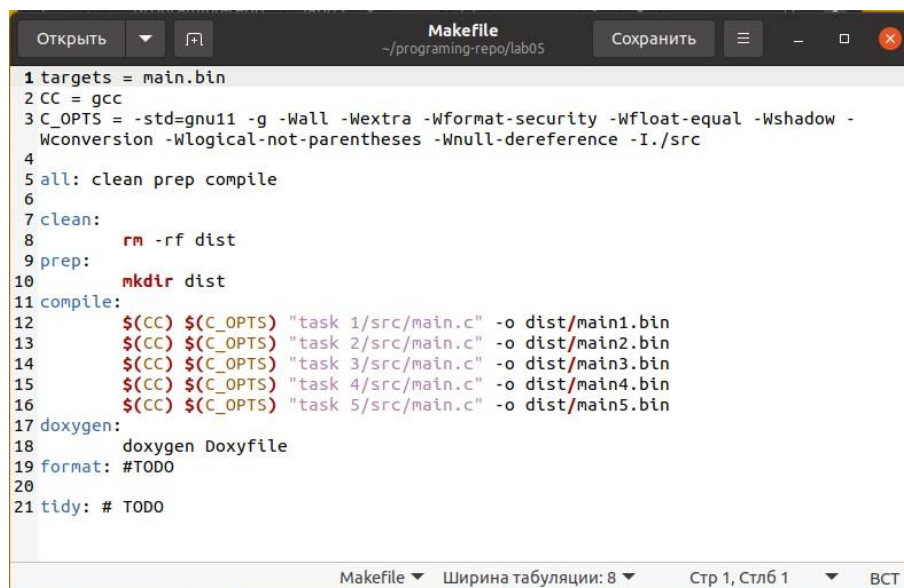
2.3 Оголошено змінні: num1, num2.



```
1 int main ()
2 {
3     int num1 = 12; // Первое заданное число
4     int num2 = 14; // Второе заданное число
5
6     // Через FOR
7     int resFor;
8     for (int i = num1; i > 0; i--) {
9         if (num1 % i == 0 && num2 % i == 0) {
10             resFor = i;
11             break;
12         }
13     }
14     // Через While
15     int resWhile;
16     int j = num1;
17     while (j > 0) {
18         if (num1 % j == 0 && num2 % j == 0) {
19             resWhile = j;
20             break;
21         }
22         j--;
23     }
24     // Через Do While
25     int resDoWhile;
26     int k = num1;
27     do {
28         if (num1 % k == 0 && num2 % k == 0) {
29             resDoWhile = k;
30             break;
31         }
32         k--;
33     } while (k > 0);
34     return 0;
35 }
36 }
```

Рисунок 2 – Реалізація програми

2.4 Внесено зміни до *Makefile* для компіляції проекту (Рис. 3).



```
1 targets = main.bin
2 CC = gcc
3 C_OPTS = -std=gnu11 -g -Wall -Wextra -Wformat-security -Wfloat-equal -Wshadow -
4 Wconversion -Wlogical-not-parentheses -Wnull-dereference -I./src
5 all: clean prep compile
6
7 clean:
8     rm -rf dist
9 prep:
10     mkdir dist
11 compile:
12     $(CC) $(C_OPTS) "task 1/src/main.c" -o dist/main1.bin
13     $(CC) $(C_OPTS) "task 2/src/main.c" -o dist/main2.bin
14     $(CC) $(C_OPTS) "task 3/src/main.c" -o dist/main3.bin
15     $(CC) $(C_OPTS) "task 4/src/main.c" -o dist/main4.bin
16     $(CC) $(C_OPTS) "task 5/src/main.c" -o dist/main5.bin
17 doxygen:
18     doxygen Doxyfile
19 format: #TODO
20
21 tidy: # TODO
```

Рисунок 3 – Структура Makefile

2.5 Проект було компільовано, використовуючи *Makefile* та запущено за допомогою *nemiver*. Програма виконалася успішно, результати завдання вказані на скриншоті (Рис.4).

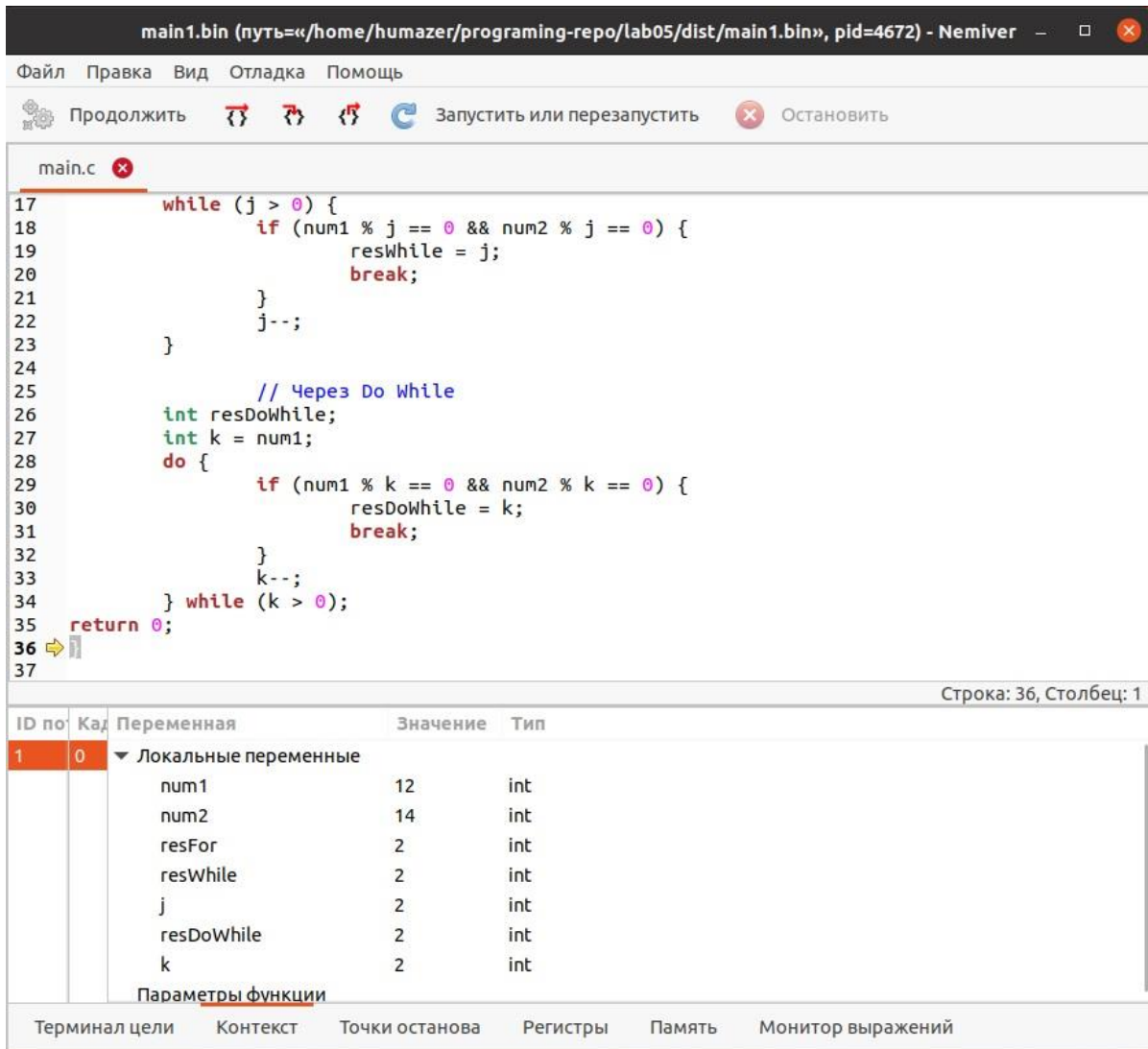


Рисунок 4 – Відлагодження програми з *nemiver* та результати завдання.

2.6 Зміни було зафіксовано та завантажено у *git*-репозиторії.

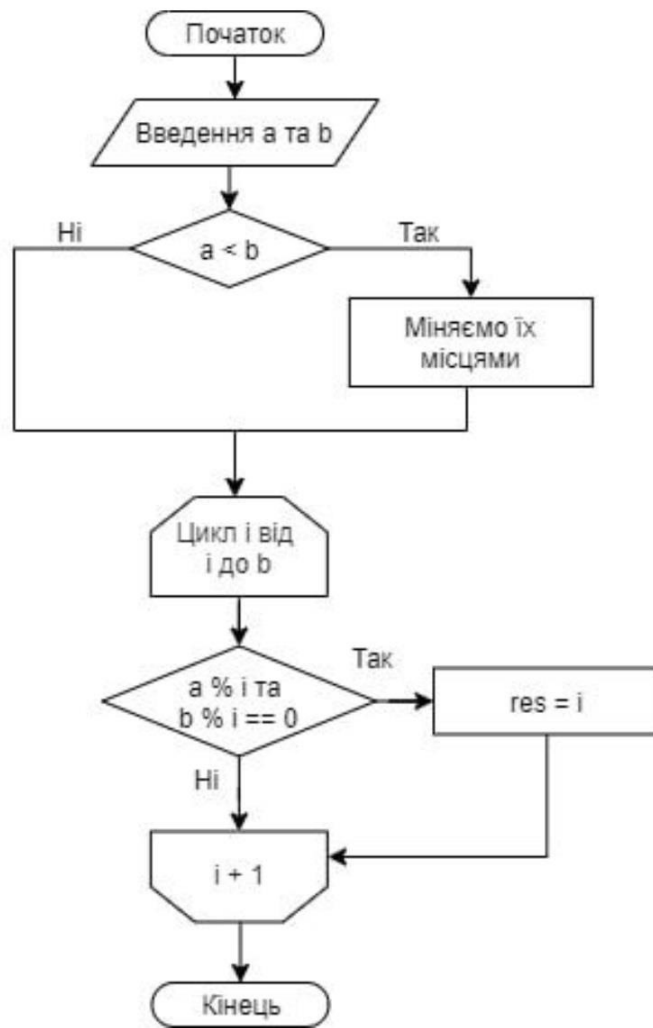


Рисунок 5 – Блок-схема

3. Висновки

При виконанні даної лабораторної роботи було набуто практичного досвіду роботи з циклічними конструкціями for, while, do while.