

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

### **ЗВІТ**

Про виконання лабораторної роботи № 5  
«Циклічні конструкції»

Керівник: викладач  
Бульба С. С.

Виконавець: студент гр. КІТ-120в  
Стегній Б. В.

Харків 2020

# Лабораторна робота № 3. Розробка програм, що розгалужуються

## 1 Вимоги

### 1.1 Розробник

- Стегній Богдан Володимирович;
- Студент групи КІТ-120в;
- 10-листопада-2020.

### 1.2 Загальне завдання

**На оцінку “відмінно”.** Необхідно виконати усі завдання з даної категорії (проте звіт та відповідні зміни до системи контролю версіями виконуються лише за одним обраним студентом варіантом).

1. Визначити найбільший спільний дільник для двох заданих чисел.
2. Визначити, чи є задане ціле число простим.
3. Визначити, чи є ціле 6-значне число «щасливим» квитком («щасливий квиток» – квиток, в якому сума першої половини чисел номера дорівнює сумі другої половини. Наприклад, білет з номером *102300* є щасливим, бо  $1 + 0 + 2 = 3 + 0 + 0$ ).
4. Визначити, чи є задане число досконалим (якщо воно дорівнює сумі своїх дільників). Наприклад, 6 - досконале число, бо  $6 = 1 + 2 + 3$ .
5. Без допомоги зовнішніх бібліотек, отримати корінь заданого числа.

### 1.3 Індивідуальне завдання

За даною формулою обчислив варіант для написання звіту

$$N_t = ((N_j - 1) \% C) + 1,$$

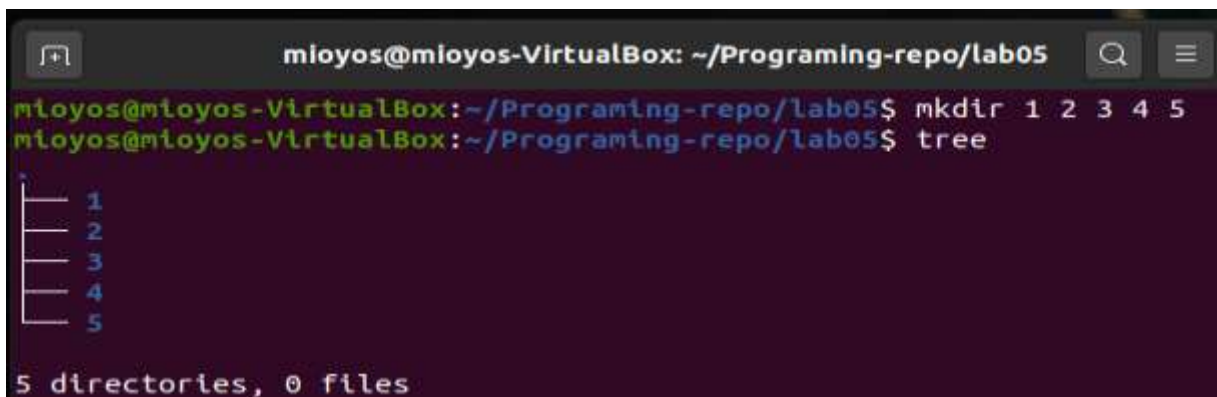
де:

- $N_j$  – номер студента у журналі групи;
- $C$  – кількість варіантів у лабораторній роботі (для розраховуємого рівня);
- $\%$  – Ділення з остачею.

1. Визначити найбільший спільний дільник для двох заданих чисел.

## 2 Виконання роботи

- 2.1 Створив директорію lab05 у репозиторії Programing-repo та директорії для завдань 1-5 за допомогою команди mkdir.



```
mloyos@mloyos-VirtualBox: ~/Programing-repo/lab05
mloyos@mloyos-VirtualBox:~/Programing-repo/lab05$ mkdir 1 2 3 4 5
mloyos@mloyos-VirtualBox:~/Programing-repo/lab05$ tree
.
├── 1
├── 2
├── 3
├── 4
└── 5

5 directories, 0 files
```

Рисунок 1 – Створення директорій

- 2.2 Заніс потрібні файли до директорій для того, щоб можна було працювати.



```
1
├── Doxyfile
├── Makefile
├── README.md
└── src
    └── main.c
```

## Рисунок 2 – потрібні файли для роботи

### 2.3 Спочатку написав код за допомогою оператора циклу for.

```
1 int main()
2 {
3     // Додаємо змінну a та задаємо її значення
4     int a = 16;
5     // Додаємо змінну b та задаємо її значення
6     int b = 4;
7     // Додаємо змінну result та задаємо її значення
8     int result = 0;
9
10
11     // Додаємо оператор циклу for, змінну nod, умову та приріст
12     for(int nod = a; nod > 0; nod--){
13         // Якщо залишки від ділення змінної a на змінну nod та ділення змінної b на
14         // змінну nod дорівнюють нулю, то результат дорівнює змінній nod
15         if(a % nod == 0 && b % nod == 0){
16             result = nod;
17             break;
18         }
19     }
```

Рисунок 3 – код для for

### 2.4 Після цього написав код для оператора циклу do..while.

```
20
21     int nod = a;
22     // Додаємо оператор циклу do..while
23     do{
24         // Якщо залишки від ділення змінної a на змінну nod та ділення змінної b на
25         // змінну nod дорівнюють нулю, то результат дорівнює змінній nod
26         if(a % nod == 0 && b % nod == 0){
27             result = nod;
28             break;
29         }
30         nod--;
31         // Умова циклу
32     }while(nod > 0);
```

Рисунок 4 – код для do..while

### 2.5 Останнім оператором циклу став while..do.

```
35     /* int nod = a */
36     // Додаємо оператор циклу while..do
37     while(nod > 0) // Умова
38     {
39         // Якщо залишки від ділення змінної a на змінну nod та ділення змінної b на
40         // змінну nod дорівнюють нулю, то результат дорівнює змінній nod
41         if(a % nod == 0 && b % nod == 0){
42             result = nod;
43             break;
44         }
45         nod--;
46     }
47     return 0;
```

Рисунок 5 – код для while..do

2.6 За допомогою команди make clean prep compile скомпілював проект.

```
mloyos@mloyos-VirtualBox:~/Programing-repo/lab05/1$ make clean prep compile
rm -rf dist
mkdir dist
gcc -std=gnu11 -g -Wall -Wextra -Wformat-security -Wfloat-equal -Wshadow -Wconv
ersion -Wlogical-not-parentheses -Wnull-dereference src/main.c -o ./dist/main.bi
n
```

Рисунок 6 – компіляція

2.7 Перевіряв працездатність програми за допомогою дебагера nemiver.

```
mloyos@mloyos-VirtualBox:~/Programing-repo/lab05/1$ nemiver ./dist/main.bin
```

Рисунок 7 – запуск дебагера

The screenshot shows the Nemiver debugger interface. The top window displays the source code of `main.c` with the following content:

```
1 int main()
2 {
3     // Додаємо змінну a та задаємо її значення
4     int a = 16;
5     // Додаємо змінну b та задаємо її значення
6     int b = 4;
7     // Додаємо змінну r та задаємо її значення
8     int result = 0;
9
10
11     // Додаємо оператор циклу for, змінну nod, умову та приріст
12     for(int nod = a; nod > 0; nod--){
13         // Якщо залишки від ділення змінної a на змінну nod та ділення змінної b на
14         if(a % nod == 0 && b % nod == 0){
15             result = nod;
16             break;
17         }
18     }
19 }
```

Below the code editor, the "Локальные переменные" (Local Variables) window is open, showing the following variables and their values:

Переменная	Значение	Тип
a	16	int
b	4	int
result	4	int
nod	4	int

The bottom of the interface shows tabs for "Терминал цели", "Контекст", "Точки останова", "Регистры", "Память", and "Монитор выражений".

Рисунок 8 – Перевірка програми на працездатність

2.8 Усі зміни були додані, зафіксовані та завантажені до онлайн-репозиторію Programing-repo за допомогою команд git add, git commit, git push.

```
mloyos@mloyos-VirtualBox:~/Programing-repo$ git add lab05/
mloyos@mloyos-VirtualBox:~/Programing-repo$ cd lab05
mloyos@mloyos-VirtualBox:~/Programing-repo/lab05$ git status
На ветке main
Ваша ветка обновлена в соответствии с «origin/main».

Изменения, которые будут включены в коммит:
(use "git restore --staged <file>..." to unstage)
    новый файл:    1/.clang-format
    новый файл:    1/.gitignore
    новый файл:    1/Doxyfile
    новый файл:    1/Makefile
    новый файл:    1/README.md
    новый файл:    1/src/main.c
    новый файл:    2/.clang-format
    новый файл:    2/.gitignore
    новый файл:    2/Doxyfile
    новый файл:    2/Makefile
    новый файл:    2/README.md
    новый файл:    2/src/main.c
    новый файл:    3/.clang-format
    новый файл:    3/.gitignore
    новый файл:    3/Doxyfile
    новый файл:    3/Makefile
    новый файл:    3/README.md
    новый файл:    3/src/main.c
    новый файл:    4/.clang-format
    новый файл:    4/.gitignore
    новый файл:    4/Doxyfile
    новый файл:    4/Makefile
    новый файл:    4/README.md
    новый файл:    4/src/main.c
    новый файл:    5/.clang-format
    новый файл:    5/.gitignore
    новый файл:    5/Doxyfile
    новый файл:    5/Makefile
    новый файл:    5/README.md
    новый файл:    5/src/main.c
```

Рисунок 9 – git add



```

mloyos@mloyos-VirtualBox:~/Programing-repo/lab05$ git commit -m "lab05"
[main de8c672] lab05
30 files changed, 3323 insertions(+)
create mode 100644 lab05/1/.clang-format
create mode 100644 lab05/1/.gitignore
create mode 100644 lab05/1/Doxyfile
create mode 100644 lab05/1/Makefile
create mode 100644 lab05/1/README.md
create mode 100644 lab05/1/src/main.c
create mode 100644 lab05/2/.clang-format
create mode 100644 lab05/2/.gitignore
create mode 100644 lab05/2/Doxyfile
create mode 100644 lab05/2/Makefile
create mode 100644 lab05/2/README.md
create mode 100644 lab05/2/src/main.c
create mode 100644 lab05/3/.clang-format
create mode 100644 lab05/3/.gitignore
create mode 100644 lab05/3/Doxyfile
create mode 100644 lab05/3/Makefile
create mode 100644 lab05/3/README.md
create mode 100644 lab05/3/src/main.c
create mode 100644 lab05/4/.clang-format
create mode 100644 lab05/4/.gitignore
create mode 100644 lab05/4/Doxyfile
create mode 100644 lab05/4/Makefile
create mode 100644 lab05/4/README.md
create mode 100644 lab05/4/src/main.c
create mode 100644 lab05/5/.clang-format
create mode 100644 lab05/5/.gitignore
create mode 100644 lab05/5/Doxyfile
create mode 100644 lab05/5/Makefile
create mode 100644 lab05/5/README.md
create mode 100644 lab05/5/src/main.c

```

Рисунок 10 – git commit

```

mloyos@mloyos-VirtualBox:~/Programing-repo/lab05$ git push
Username for 'https://github.com': mloyosss
Password for 'https://mloyosss@github.com':
Перечисление объектов: 28, готово.
Подсчет объектов: 100% (28/28), готово.
Сжатие объектов: 100% (21/21), готово.
Запись объектов: 100% (27/27), 8.88 KiB | 1.11 MiB/s, готово.
Total 27 (delta 5), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (5/5), completed with 1 local object.
To https://github.com/mloyosss/Programing-repo
  901eaaf..de8c672  main -> main
mloyos@mloyos-VirtualBox:~/Programing-repo/lab05$

```

Рисунок 11 – git push

## 2.9 Повний код програми.

```
1 int main()
2
3     // Додаємо змінну a та задаємо її значення
4     int a = 16;
5     // Додаємо змінну b та задаємо її значення
6     int b = 4;
7     // Додаємо змінну r та задаємо її значення
8     int result = 0;
9
10
11     // Додаємо оператор циклу for, змінну nod, умову та приріст
12     for(int nod = a; nod > 0; nod--){
13         // Якщо залишки від ділення змінної a на змінну nod та ділення змінної b на
змінну nod дорівнюють нулю, то результат дорівнює змінній nod
14         if(a % nod == 0 && b % nod == 0){
15             result = nod;
16             break;
17         }
18     }
19
20
21     int nod = a;
22     // Додаємо оператор циклу do..while
23     do{
24         // Якщо залишки від ділення змінної a на змінну nod та ділення змінної b на
змінну nod дорівнюють нулю, то результат дорівнює змінній nod
25         if(a % nod == 0 && b % nod == 0){
26             result = nod;
27             break;
28         }
29         nod--;
30         // Умова циклу
31     }while(nod > 0);
32
33
34
35     /* int nod = a */
36     // Додаємо оператор циклу while..do
37     while(nod > 0) // Умова
38     {
39         // Якщо залишки від ділення змінної a на змінну nod та ділення змінної b на
змінну nod дорівнюють нулю, то результат дорівнює змінній nod
40         if(a % nod == 0 && b % nod == 0){
41             result = nod;
42             break;
43         }
44         nod--;
45     }
46
47     return 0;
48
```

Рисунок 12 – код



**2.10** Зробив блок-схему програми.

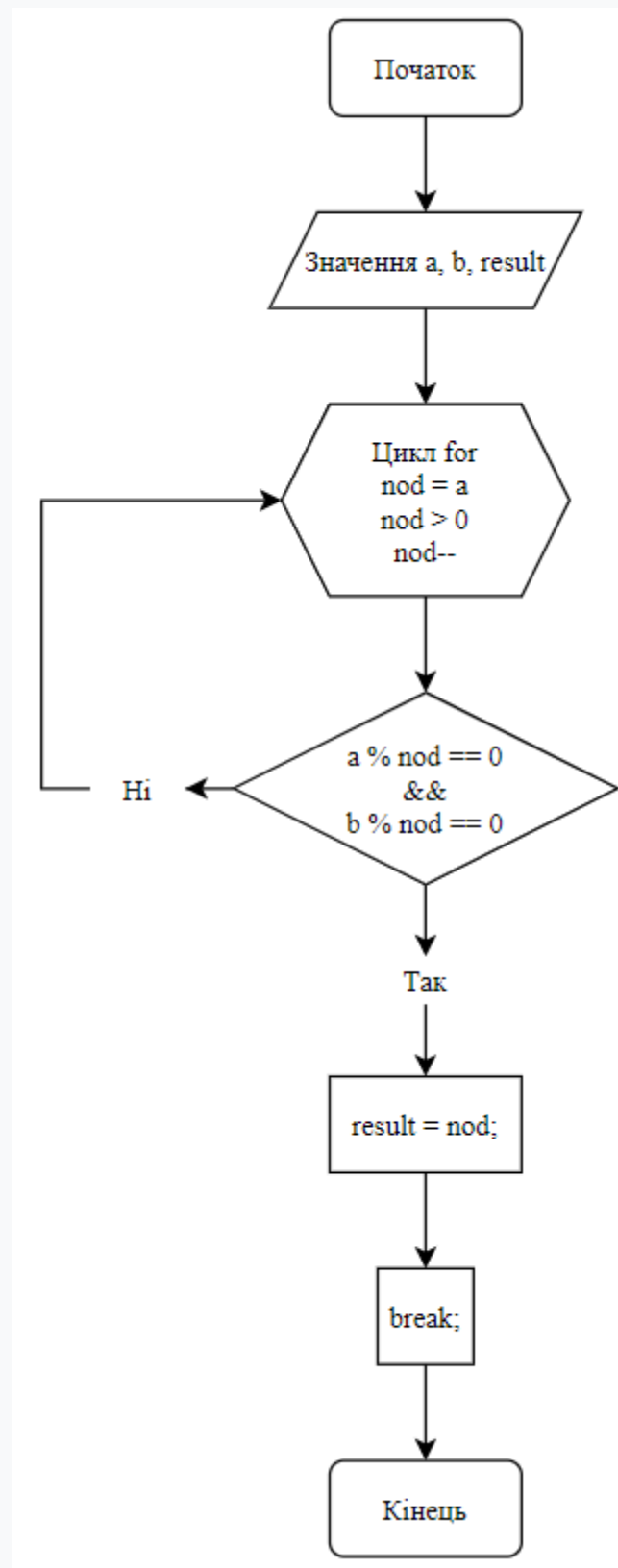


Рисунок 13 – блок-схема

## Висновки

При виконанні лабораторної роботи № 5 «Циклічні конструкції», розробив 5 програм. Розробив програму, яка може визначати найбільший спільний дільник серед двох чисел. Були розвинуті нові навички у роботі з середовищем Linux.

