

Міністерство 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  
Бабенко А. П.

Харків 2020

# Лабораторна робота №5. Циклічні конструкції

## 1. Вимоги

### 1.1 Розробник

- Бабенко Антон Павлович;
- Студент групи КІТ-120в;
- 29 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

Розробити програму за допомогою трьох типів циклів: *for*, *while-do*, *do-while* :

Визначити найбільший спільний дільник для двох заданих чисел.

## 2 Виконання роботи

2.1 Створення директорії lab05. Зображено на рис.1.

```
anton@anton-X55VD:~/dev/Programming-repo$ mkdir lab05
```

Рисунок 1 - створення директорії

2.2 Створення файлу з вихідним кодом. Зображено на рис.2.

```
anton@anton-X55VD:~/dev/Programming-repo/lab05$ mkdir src
anton@anton-X55VD:~/dev/Programming-repo/lab05$ cd src/
anton@anton-X55VD:~/dev/Programming-repo/lab05/src$ touch task1.c
```

Рисунок 2 — створення файлу з кодом

2.3 Запуск редактора коду. Зображено на рис.3.

```
anton@anton-X55VD:~/dev/Programming-repo/lab05/src$ kate task1.c
```

Рисунок 3 — запуск редактора

2.4 Написання коду і коментарів до нього. Зображено на рис.4.

```
int main()
{
    // Объявляем переменные для записи чисел и результата
    int a = 17;
    int b = 13;
    int res;
    int i = 1;

    // Запись младшего числа в переменную b
    if (a < b)
    {
        int x = a;
        a = b;
        b = x;
    }

    // Цикл прохождения от 1 до младшего числа
    for (i = 1; i <= b; i++)
    {
        // Если остаток от деления двух чисел == 0, записываем его в переменную
        if ((a % i) == 0 && (b % i) == 0)
        {
            res = i;
        }
    }

    return 0;
}
```

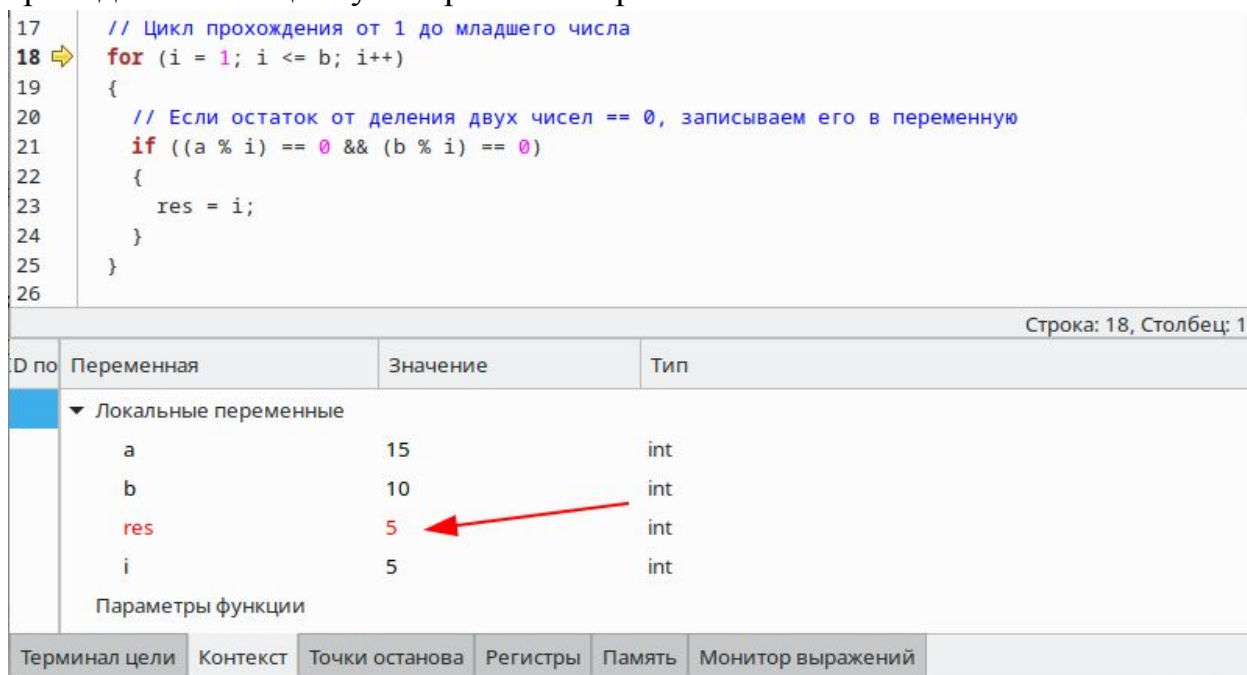
Рисунок 4 — код програми

2.5 Компіляція проекту за допомогою команди “make clean prep compile”. Зображено на рис.5.

```
anton@anton-X55VD:~/dev/Programming-repo/lab05$ 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 -I./src src/task1.c -o ./dist/task1.bin
```

Рисунок 5 — компіляція проекту

2.6 Відкрито у відлагоднику nemiver виконуючий файл main.bin. Ставимо точку зупину, проходимо по файлу і бачимо зміну значення змінної в момент проходження по циклу. Зображено на рис.6.



```
17 // Цикл прохождение от 1 до младшего числа
18 ➔ for (i = 1; i <= b; i++)
19 {
20     // Если остаток от деления двух чисел == 0, записываем его в переменную
21     if ((a % i) == 0 && (b % i) == 0)
22     {
23         res = i;
24     }
25 }
26
```

Строка: 18, Столбец: 1

Имя	Значение	Тип
Локальные переменные		
a	15	int
b	10	int
res	5	int
i	5	int
Параметры функции		

Терминал цели | Контекст | Точки останова | Регистры | Память | Монитор выражений

Рисунок 6 — файл у відлагоднику

2.9 Створення блок-схеми програми. Зображення блок-схеми на рис.7.

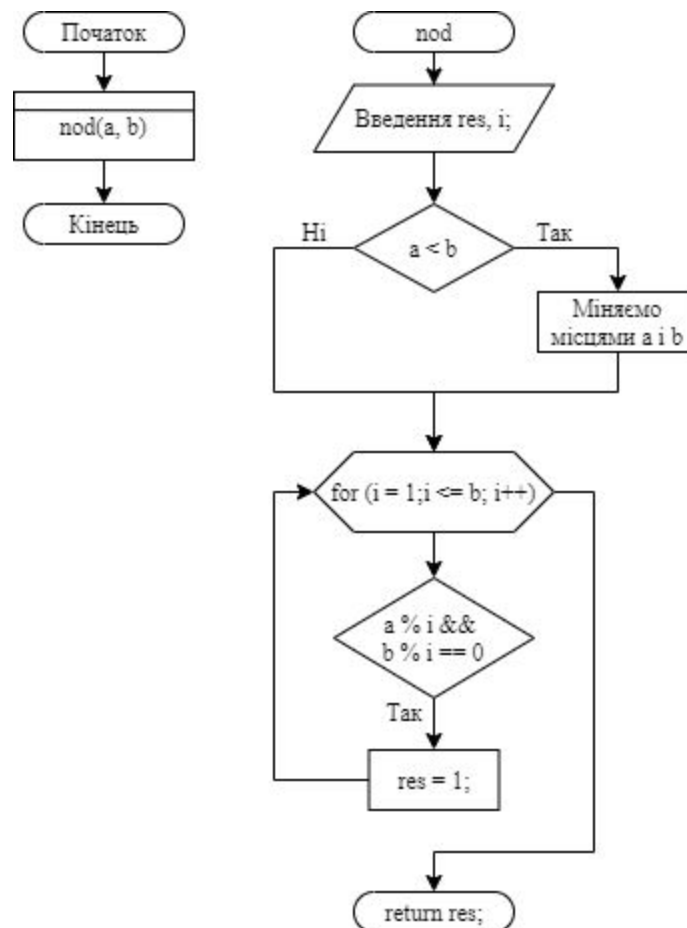


Рисунок 7 — блок-схема

2.10 Виконання опису функцій. Зображення опису зі сторінок документації doxygen на рис. 8, 9, 10.

### Опис функцій

#### ◆ get\_rand()

int get\_rand ( )

Отримання псевдовипадкового числа

Функція, що генерує і повертає псевдовипадкове число

**Повертає**

Псевдовипадкове число (random)

### ◆ nod()

```
int nod ( int a,  
          int b  
          )
```

Визначення найбільшого спільного дільника

Функція, що визначає найбільший спільний дільник для двох заданих чисел

#### Аргументи

**a** одне з чисел для знаходження нсд

**b** псевдовипадкове число для знаходження нсд

#### Повертає

Результат роботи функції (res)

### ◆ main()

```
int main ( )
```

Головна функція

Послідовність дій:

- виклик функції srand
- виклик функції nod

#### Повертає

успішний код виконання програми (0)

Рисунок 8, 9, 10 — опис функцій

## Висновки

При виконанні даної лабораторної роботи було набуто навичок розробки програм з циклічними конструкціями, а також створено програму, яка визначає найбільший спільний дільник двох чисел.

