

# Лабораторна робота №5.

## Циклічні конструкції

### 1.1 Розробник

- Бобченко Дмитро Олександрович
- студента групи КІТ-320
- 29.11.2020

### 1.2 Загальне завдання

Необхідно виконати усі завдання з даної категорії (проте звіт виконується лише за варіантів, обчислений за раніше-визначеною формулою).

### 1.3 Індивідуальне завдання

Визначити, чи є задане число досконалим (якщо воно дорівнює сумі своїх дільників). Наприклад, 6 - досконале число, бо  $6 = 1 + 2 + 3$ .

## 2.Опис програми

### 2.1 Визначити, чи є задане число досконалим

### 2.2 Код програми

```
#define NUM 28 // заданое число

int main () {
    int sum = 0;
    int temp_for = 0;
    int result_for = 0;
    for (int i = 1; i < NUM; i++) { // делитель на цело с остатком 0
        temp_for = NUM % i;
        if (temp_for == 0) { // сумируем если остаток 0
            sum += i;
        }
    }
    if (sum == NUM) {
        result_for = 1; // число совершенное
    }
    else {
        result_for = 0; //число не совершенное
    }
}
```

```

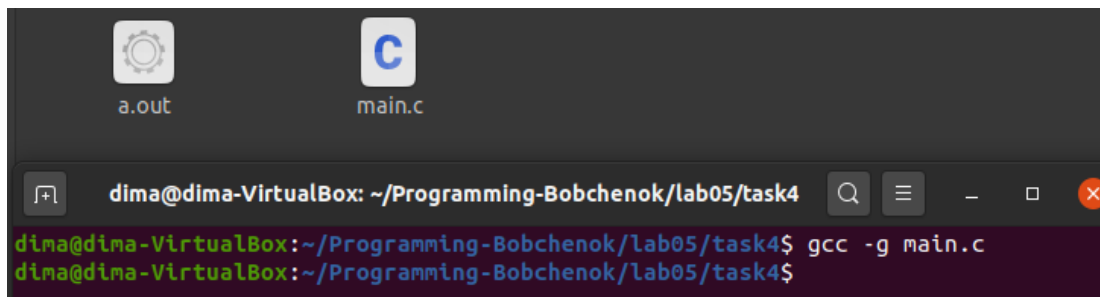
int sum_while = 0;
int temp_while = 0;
int result_while = 0;
int step_while = 1; // деление числа
while (step_while < NUM) {
    temp_while = NUM % step_while;
    if (temp_while == 0) { // суммируем если остаток 0
    }
    step_while++;
}
if (sum == NUM) {
    result_while = 1; // число совершенное
}
else {
    result_while = 0; // число не совершенное
}

int sum_do = 0;
int temp_do = 0;
int result_do = 0;
int step_do = 1;
do {
    temp_do = NUM % step_do;
    if (temp_do == 0) {
        sum_do += step_do;
    }
    step_do++;
} while (step_do < NUM);
if (sum == NUM) {
    result_do = 1; //число совершенное
}
else {
    result_do = 0; //число не совершенное
}

return 0;
}

```

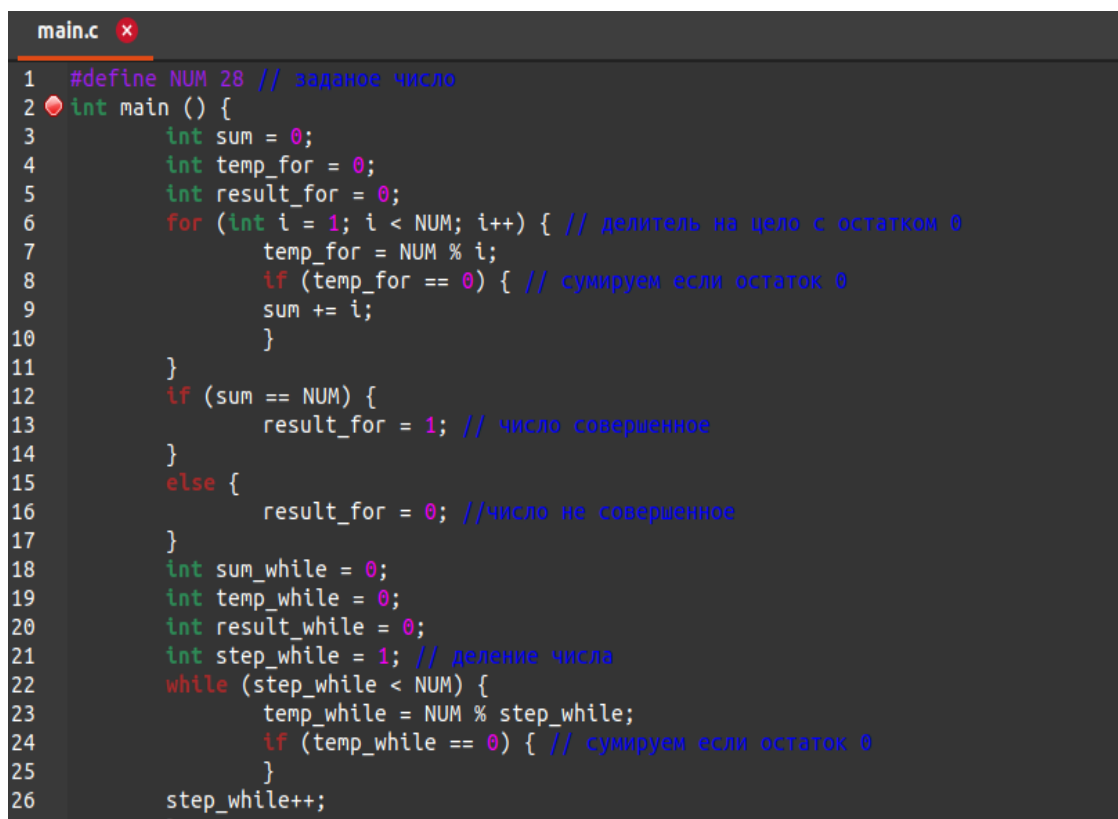
## 2.3 Компіляція програми



The screenshot shows a terminal window with a dark background. At the top, there are two icons: a gear icon labeled 'a.out' and a blue 'C' icon labeled 'main.c'. The terminal prompt is 'dima@dima-VirtualBox: ~/Programming-Bobchenok/lab05/task4'. The user has entered the command 'gcc -g main.c' and the terminal has responded with the same prompt, indicating the command has been executed.

```
dima@dima-VirtualBox: ~/Programming-Bobchenok/lab05/task4
dima@dima-VirtualBox:~/Programming-Bobchenok/lab05/task4$ gcc -g main.c
dima@dima-VirtualBox:~/Programming-Bobchenok/lab05/task4$
```

## 2.4 Відлагодження програми за допомогою nemiver



The screenshot shows a code editor window titled 'main.c'. The code is written in C and is a program to check if a number is perfect. It defines a constant NUM as 28 and implements a function main that uses a for loop to calculate the sum of divisors and a while loop to calculate the sum of the number's digits. Comments in Russian are present throughout the code.

```
1 #define NUM 28 // заданное число
2 int main () {
3     int sum = 0;
4     int temp_for = 0;
5     int result_for = 0;
6     for (int i = 1; i < NUM; i++) { // делитель на цело с остатком 0
7         temp_for = NUM % i;
8         if (temp_for == 0) { // сумируем если остаток 0
9             sum += i;
10        }
11    }
12    if (sum == NUM) {
13        result_for = 1; // число совершенное
14    }
15    else {
16        result_for = 0; //число не совершенное
17    }
18    int sum_while = 0;
19    int temp_while = 0;
20    int result_while = 0;
21    int step_while = 1; // деление числа
22    while (step_while < NUM) {
23        temp_while = NUM % step_while;
24        if (temp_while == 0) { // сумируем если остаток 0
25        }
26        step_while++;
27    }
```

```

main.c x
28     if (sum == NUM) {
29         result_while = 1; // число совершенное
30     }
31     else {
32         result_while = 0; // число не совершенное
33     }
34
35     int sum_do = 0;
36     int temp_do = 0;
37     int result_do = 0;
38     int step_do = 1;
39     do {
40         temp_do = NUM % step_do;
41         if (temp_do == 0) {
42             sum_do += step_do;
43         }
44         step_do++;
45     } while (step_do < NUM);
46     if (sum == NUM) {
47         result_do = 1; //число совершенное
48     }
49     else {
50         result_do = 0; //число не совершенное
51     }
52
53     return 0;
54 }

```

Thread	Variable	Value	Type
Local Variables			
	sum	28	int
	temp_for	1	int
	result_for	1	int
	sum_while	0	int
	temp_while	1	int
	result_while	1	int
	step_while	28	int
	sum_do	28	int
	temp_do	1	int
	result_do	1	int
	step_do	28	int
Function Arguments			
Target Terminal Context Breakpoints Registers Memory Expression Monitor			

### **3 Варіанти використання**

За допомогою цієї програми ми можемо розрахувати чи є доскональне число або ні

### **4 Висновок**

У ході виконанні завдання ми змогли реалізувати за допомогою циклів яка завдяки їм перевірює чи є задане число доскональним або ні