

Лабораторна робота № 5. Циклічні структури

1 ВИМОГИ

1.1 Розробник

- Пташник Максим Андрійович;
- студент групи КІТ-121Г;
- 23-гру-2021.

1.2 Загальне завдання

Розробити програму використовуючи функції на мові С.

1.3 Індивідуальне завдання

Розробити функцію, яка буде перевіряти число на досконалість.

2 ОПИС ПРОГРАМИ

2.1 Функціональне призначення

Програма призначена для перевірки числа на досконалість, шляхом знаходження суми всіх його дільників, крім нього самого.

При перевірці числа діють наступні обмеження:

- ▢ числа які передаються в функцію повинні бути *int*.
- ▢ результат зберігається у змінній *result*.
- ▢ подивитися результат виконання можна тільки в режимі відлагодження.

2.2 Опис логічної структури

Програма бере число і передає його в цикл. В циклі програма знаходить дільники числа через ділення по модулю на «*i*», де «*i*» - ціле число, починаючи з одиниці (Оскільки на 0 ділити не можна). Далі програма складає всі числа «*i*», які є дільниками. Якщо сума дільників дорівнює самому числу, програма повертає 1, якщо ні – 0.

2.2.1 Функція перевірки на доскональність

get_perfect_number(int x)

Призначення: перевірка числа на досконалість.

Опис роботи: в циклі знаходить дільники числа через ділення по модулю на «i», де «i» - ціле число, починаючи з одиниці (Оскільки на 0 ділити не можна). Далі складає всі числа «i», які є дільниками. Якщо сума дільників дорівнює самому числу, програма повертає 1, якщо ні – 0.

2.2.2 Основна функція

*int main(int argc, char** argv)*

Призначення: головна функція.

Опис роботи:

- якщо через аргументи командної строки нічого не передається, то в аргумент x заноситься випадкове число.
- число, яке було передано через аргумент строки або випадкове, передається в функцію *get_perfect_number*.
- результат виконання записується в змінну *result*.

2.2.3 Структура проекту

lab05

```
└── doc
    ├── lab05.docx
    ├── lab05.md
    └── lab05.pdf
```

2.3 Важливі фрагменти програми

```
int get_perfect_number( int x){
    int sum = 0;
    //Знаходимо дільники.
    for(int i = 1; i < x; i++) {
        if ( x % i == 0){
            sum += i;
        }
    }
    // якщо сума дільників дорівнює числу x, то число x - досконале.
    if ( x == sum){
        return 1;
    }
    // в іншому випадку, число не є досконалим.
    else {
        return 0;
    }
}
```

ВИКОРИСТАННЯ

Для демонстрації результатів задачі використовується:

- покрокове виконання програми в утиліті lldb;

3. Варіант використання: послідовність дій для запуску програми у режимі відлагодження:

- запустити програму у відлагоднику lldb;
- поставити точку зупинки на функції main (строка з return 0;);
- запустити програму;
- подивитись результати виконання програми, оглядаючи значення змінної result;

Process 16223 stopped

* thread #1, name = 'main.bin', stop reason = breakpoint 1.1

frame #0: 0x000000000040124f main.bin`main(argc=1, argv=0x00007ffffffdf48)

at main.c:77:9

```
74         result = get_perfect_number((int)strtod(argv[1], 0));
75     }
76
-> 77         return 0;
78     }
79
80
```

(lldb) p result

(int) \$0 = 0

(lldb)

ВИСНОВКИ

При виконанні даної лабораторної роботи було набуто практичного досвіду роботи зі створення функціональної програми, яка перевіряє чи є чимло досконалим.