

Лабораторна робота № 5. Циклічні структури

1 Вимоги

1.1 Розробник

- Силка Михайло Миколайович;
- студент групи КІТ-121Г;
- 18-гру-2021.

1.2 Загальне завдання

Переробити лінійну програмуу функціональну

1.3 Індивідуальне завдання

Сформувати функцію, яка буде визначати тип числа: просте чин ні

2 Опис програми

2.1 Функціональне призначення

Програма призначеня для визначення типу числа

При визначенні числа діють наступні обмеження:

- числа які передаються в функцію повинні бути int
- число повинно бути відмінним від 0

Результат зберігається у змінній tf.

Подивитися результат виконання можна тільки в дебагері

2.2 Опис логічної структури

Програма ділить всі числа що входять в задане по модулю на два до тих пір поки не знайдеться число, в якого результат від ділення по модулю буде дорівнювати нулю або доки не закінчатся числа

2.2.1 Функція знаходження найбільшого загального дільника

```
int cycle_for (int x)
```

Призначення: повертання результату визначення типу числа переданого як аргумент

Опис роботи: в циклі обмеженому переданим числом ділить всі числа по модулю на 2, доки не знайдеться число, в якого результат від ділення по модулю буде дорівнювати нулю або доки не закінчатся числа

2.2.2 Основна функція

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

Призначення: головна функція.

Опис роботи:

- якщо через аргументи командоні строки нічого не передається, то для змінної x створюється випадкове число;
- число, яке було передано через аргументи строки або випадкові, передаються в функцію `int cycle_for (int x);`
- результат виконання записується в змінну `tf`.

2.2.3 Структура проекту

lab05

```

├── Doxyfile
├── Makefile
├── doc
│   ├── rep_lab05.docx
│   ├── rep_lab05.md
│   └── rep_lab05.pdf
└── src
    └── main.c

```

2.3 Важливі фрагменти програми

Визначення числа

```

if (argc == 1){
    srand(time(NULL));
    x = rand()%100;
    tf = cycle_for(x);
} else {
    tf = cycle_for((int)strtod(argv[1], 0));
}

```

Визначення типу числа

```

if (x != 0){
    for (int i = 0; i < x; i++){
        if (i == 0){
            if (x != 2 && x % 2 == 0){
                tf = 1;
                break;
            } else {
                continue;
            }
        }
    }
}

```

```

        }
    }

    if (i == 1){
        continue;
    }

    if (x % i == 0){
        tf = 1;
        break;
    }
}
}

```

Варіант використання

Для демонстрації результатів задачі використовується:

- покрокове виконання програми в утиліті lldb;

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 tf

Process 4494 stopped

(lldb) r

Process	6158	launched:
'/home/yuzo/lab/programing-Sylka/lab08/lab07/lab05/src/main.bin' (x86_64)		

Process 6158 stopped

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

frame #0: 0x00000000004011cf main.bin`main(argc=1, argv=0x00007ffffffdf18)

at main.c:53:5

```
50          tf = cycle_for((int)strtod(argv[1], 0));  
51      }  
52  
-> 53      return (0);  
54  }  
55  
56
```

(lldb) fr v

(int) argc = 1

(char **) argv = 0x00007ffffffdf18

(int) x = 99

(int) tf = 1

Висновки:

При виконанні даної лабораторної роботи я покращив свої навички роботи з циклами та функціями