

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ



**Дніпровський національний університет
залізничного транспорту імені академіка В. Лазаряна**

Кафедра «Комп'ютерні інформаційні технології»

Лабораторна робота №4

з дисципліни «Об'єктно-орієнтоване програмування»

на тему: «Функціональні об'єкти»

Виконав:
студент гр.ПЗ1911
Сафонов Д.Є.
Прийняла:
Демидович І. М.

Дніпро, 2020

1 Постановка задачі згідно загального та індивідуального завдання.

Варіант	Об'єкт
6	На вход подается массив. «Состояние» хранит целое число. Преобразовать массив в другой массив, в котором сначала располагаются те элементы, который делятся на значения «состояния» без остатка, затем те, который делятся на значение состояния, уменьшенное на 1, потом на 2 и т.д. до 0.

2 Текст програми.

“main.cpp”

```
#include <iostream>
#include "functor.h"

template <class T>
void print_vec(std::vector<T>& v){
    for (auto& c : v)
        std::cout << c << " ";
    std::cout << "\n";
}

int main() {
    Functor f(7);
    std::vector v{15, 24, 56, 66};
    std::cout << "state = " << f.state << "\n";

    std::cout << "presort\n";

    print_vec(v);

    f(v);
    // 15, 24, 56, 66, state = 7
    // 7: 56
    // 6: 24, 66 -> 4: 24, 3: 66
    // 5: 15
    // 56, 24, 66, 15

    std::cout << "postort\n";
    print_vec(v);
}
```

“functor.h”

```
#ifndef FUNCTOR_H
#define FUNCTOR_H

#include <vector>
#include <algorithm>

template<class T> //T positive integral type
class Functor {
public:
    Functor(T state) : state{state} {}
    T state;
    void operator()(std::vector<T>& vec) {
        std::sort(vec.begin(), vec.end(), [&](const T& a, const T& b) {
            auto cmp = a <=> a; //std::compare_three_way_result_t<T> cant be uninitialized
            for (T state = this->state; cmp == 0 && state >= 0; state--)
                cmp = (gcd(a, state) <=> gcd(b, state));
            return cmp > 0;
        });
    }
private:
    int gcd(T a, T b) {
        T t;
        while (b != 0) {
            t = b;
            b = a % b;
            a = t;
        }
        return a;
    }
};
#endif
```

3 Результати виконання.

```
state = 7  
presort  
15 24 56 66  
postort  
56 24 66 15  
Press <RETURN> to close this window...  
□
```

Рисунок 1

56 єдине число у масиві, яке ділиться на 7. 24 та 66 діляться на 6, обидва не діляться на 5, 24 ділиться на 4, а 66 — ні, тож 24 буде йти раніше. 15 залишилося останнім.

4 Висновок.

Функтори обов'язкові у ОО мовах програмування.

Також вони необхідні для можливості передавати функції як параметри, але у більшості мов ця роль вже виконана. Наприклад лямбда функціями(c++, python), або звичайними функціями(python), або вказівниками на функцію(c++, asm8086).

Авжеж іноді потрібно викликати функцію повторюючи один з аргументів, але можна не робити для цього клас, та створювати його об'єкт, а просто зберегти аргумент який повторюється.