

МИНОБРНАУКИ РОССИИ
САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ
ЭЛЕКТРОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ
«ЛЭТИ» ИМ. В.И. УЛЬЯНОВА (ЛЕНИНА)
Кафедра МО ЭВМ

ОТЧЕТ
по лабораторной работе №1
по дисциплине «Алгоритмы и структуры данных»
Тема: Рекурсия

Студент гр. 9382

Демин В.В.

Преподаватель

Фирсов М.А.

Санкт-Петербург

2020

Цель работы.

Ознакомиться с основными понятиями и приёмами рекурсивного программирования, получить навыки программирования рекурсивных процедур и функций на языке программирования C++.

Задание.

Вариант №5

Функция $f(n)$ определена для целых положительных чисел:

$$f(n) = \begin{cases} 1, & \text{если } n = 1, \\ \sum_{i=2}^n f(n \operatorname{div} i), & \text{если } n \geq 2. \end{cases}$$

Вычислить $f(k)$ для $k = 15, 16, \dots, 30$.

Ход работа.

В процессе рекурсии рассматривается функции которая при принятом значении $n = 1$ принимает значение 1, при $n \geq 2$ значение представляет себя сумму точек функции от $n \operatorname{div} i$ ($i=2$ to n).

Программа написана в рекурентном стиле, так как функция при $n \geq 2$ вызывает значения функции от других значений.

При интерактивном решении задачи для каждой отдельной точки n , необходимо писать отдельную программу, так как для каждого значения, вызываются разные функции для сумму. $n \operatorname{div} i$ ($i=2$ to n) при разных n будет иметь абсолютно разные значения.

Функции и структуры данных.

Структуры данных в данной программе не используются. Функции `type_selection()` и `print_result()` необходимы для реализации программы. Функция $f(\text{int } n)$ является рекурсивной функцией, которая представляет из себя сумму точек функции $n \operatorname{div} i$ ($i=2$ to n).

Выводы.

В данной задаче использовать рекурентное решение является наиболее грамотным решением, так как сама функция является сумму других значений функции.

Тестирование.

№ п/п	Входные данные	Выходные данные	Комментарии
1.	Hello	Incorrect data	
2.	-1	Incorrect data	
3.	0	Incorrect data	
4.	5	f(5) : 4	
5.	4	f(4) : 3	
6.	3	f(3) : 2	
7.	40	f(40) : 137	

ПРИЛОЖЕНИЕ А

ИСХОДНЫЙ КОД ПРОГРАММЫ

Название файла: main.cpp

```
#include <iostream>
#include <fstream>

//количество отступов для вывода рекурсивных функций
static int k = 0;
//для вывода промежуточных значений
static std::ofstream fout;

//recursive function
long f(long n) {
    //начало функции f(n)
    fout << "startf: f(" << n << ")::\\";
    fout << "\n";
    long sumFunctionValues = 0;
    if (n == 1) { //f(1)=1
        fout << "\n";
        for (int i = 0; i < k; i++) {
            fout << "\t";
        }
        fout << "/endf f(" << n << ")::" << 1;
        fout << "\n";

        return 1;
    }
    if (n >= 2) {
        for (long i = 2; i <= n; i++) {
            //вывод отступов
            k++;
            for (int i = 0; i < k; i++) {
                fout << "\t";
            }
            // сумма значений функции
            sumFunctionValues += f(n / i);

            k--;
        }
    }
}
```

```

        fout << "\n";
    }
    //ВЫВОД ОТСТУПОВ
    for (int i = 1; i <= k; i++) {
        fout << "\t";
    }
    //конец функции f(n)
    fout << "/endif /f(" << n << ")::" << sumFunctionValues;
    fout << "\n";

    return sumFunctionValues;
}
return 0;
}

//функция вывода рекурсивной функции от значений на промежутке
[a,b]
//либо в файл, либо в консоль
void print_result(int a, int b) {
    long n = 0;
    int type = 0;
    std::cout << "Print to console - 1\n";
    std::cout << "Print to file - 2\n";
    std::cin >> type;

    std::ofstream file;
    std::string name_f;
    if (type == 2) {
        std::cout << "Enter file name\n";
        std::cin >> name_f;
        file.open(name_f);
    }
    for (int i = a; i < b; ++i) {
        fout << "f(" << i << ") : " << "\n";
        fout << "-----"
-----" << "\n";
        n = f(i);
        if (type == 1) {

```

```

        std::cout << "f(" << i << ") : " << n << "\n";
    } else {

        if (file.is_open()) {
            file << "f(" << i << ") : " << n << "\n";
        } else {
            std::cout<<"fileIsClose";
        }
    }
}

//type_selection
void type_selection() {
    //тип вызова рекурсивной функции
    int type = 0;
    // значение для вызова рекурсивной функции
    int value = 0;

    std::ifstream file;
    std::string name_f;

    std::cout << "select type:\n 1-one value from console,2 - from
15 to 30,3 - one value from file\n";
    std::cin >> type;//ввод типа
    switch (type) {
        case 1:
            std::cout << "enter value\n";
            std::cin >> value;
            if (value < 1) { //проверка на ошибку
                std::cout << "Incorrect elem";
                break;
            }
            //вызов функции которая выведет на экран значение
            рекурсивной функции от value
            print_result(value, value + 1);
            break;
        case 2:

```

```

        //вызов функции которая выведет на экран значения
рекурсивной функции от промежутка от [15,30]
        print_result(15, 31);
        break;
    case 3:
        std::cout << "Enter file name\n";
        std::cin >> name_f;
        file.open(name_f);
        file>>value;
        if(file.is_open()){
            print_result(value,value+1);
        } else{
            std::cout<<"fileIsClose";
        }
        file.close();
        break;
    default:
        std::cout << "error: wrong type";
    }
    std::cout << "intermediate values are written to the file
intermediateResults.txt";
}

int main() {
    //открытие файла для промежуточных значений
    fout.open("intermediateResults.txt");
    //выбор типа для вызова функции
    type_selection();
    //закрытие файла для промежуточных значений
    fout.close();
    return 0;
}

```