

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. 9383

Корсунов А.А.

Преподаватель

Попова Е.В.

Санкт-Петербург

2020

Цель работы.

Познакомиться с основными понятиями и приёмами рекурсивного программирования и написать программу на языке программирования C++.

Основные теоретические положения.

Рекурсивным называют алгоритм, в описании которого прямо или косвенно содержится обращение к самому себе. Рекурсивная функция — это функция, которая вызывает саму себя.

Задание.

18. Пусть определена функция Φ преобразования целочисленного вектора α :

$$\Phi(\alpha) = \begin{cases} \alpha, & \text{если } \|\alpha\| = 1, \\ ab, & \text{если } \|\alpha\| = 2, \alpha = ab \text{ и } a \leq b, \\ ba, & \text{если } \|\alpha\| = 2, \alpha = ab \text{ и } b < a, \\ \Phi(\beta)\Phi(\gamma), & \text{если } \|\alpha\| > 2, \alpha = \beta\gamma, \text{ где } \|\beta\| = \|\gamma\| \text{ или } \|\beta\| = \|\gamma\| + 1. \end{cases}$$

Например: $\Phi(1,2,3,4,5) = 1,2,3,4,5$; $\Phi(4,3,2,1) = 3,4,1,2$; $\Phi(4,3,2) = 3,4,2$. Отметим, что функция Φ преобразует вектор, не меняя его длину. Реализовать функцию Φ рекурсивно.

Ход работы.

Для поставленной задачи была разработана программа с использованием функции `fix(int ivector[], int start, int finish)`, которая рекурсивно меняет вектор, поданный на вход программе. Стоит подробнее пояснить о работе, как самой функции, так и всей программы в целом.

1) функция `fix(int ivector[], int start, int finish)`:

Данная функция принимает на вход три аргумента: а) массив из значений векторов, б) индекс первого элемента вектора, в) индекс последнего элемента вектора.

Первое условие `if` проверяет длину вектора — если она ≤ 1 , то происходит выход из функции. Следующий `if` считает разницу между элементами с

индексами `start` и `finish`: если разница меньше двух и первый элемент больше второго (то есть, элементы являются «соседями»), то функция меняет их местами и происходит выход из функции. Если разница ≥ 2 , то происходит вызов функции `fix` два раза с измененным значением `finish` в первом вызове и измененным значением `start` во втором (таким образом массив «условно» делится на две части, первая из которых больше (на единицу) или равна второй вместе с измененными значениями индексов на первый и последний элементы).

2) функция `main()`:

В начале функции вызывается функция `setlocale()`, для того, чтобы в консоли корректно отображалась кириллица. Далее происходит инициализация необходимых переменных: `ivector[100]` – массив для значений вектора с длиной 100, `key` - «ключ» для проверки вводимого значения и записи его в массив, `icount` – переменная, которая считает количество вводимых переменных. Следует учесть, что ввод будет продолжаться, пока не встретит символ латиницы (или любой другой символ, который нельзя привести к типу `int`). После заполнения массива значениями вектора, вызывается функция `fix()`, которая меняет значения вектора согласно заданию и выводит его на экран.

Пример работы программы.

№ п/п	Входные данные	Выходные данные	Комментарии
1	Значения — 4, 3, 2, 1	{3, 4, 1, 2}	Значения вводятся через знак перевода строки
2	Значения — 9, 2, 6	{2, 9, 6}	
3	Значения — 8	{8}	

Иллюстрация работы программы.

*IDE – Code::Blocks 20.03

```

Введите значения вектора
4
3
2
1
!
{3, 4, 1, 2}
Process returned 0 (0x0)   execution time : 7.341 s
Press any key to continue.

```

Рисунок 1 - Пример работы программы с входными данными №1

```

Введите значения вектора
9
2
6
!
{2, 9, 6}
Process returned 0 (0x0)   execution time : 4.255 s
Press any key to continue.

```

Рисунок 2 - Пример работы программы с входными данными №2

```

Введите значения вектора
8
!
{8}
Process returned 0 (0x0)   execution time : 2.371 s
Press any key to continue.

```

Рисунок 3 - Пример работы программы с входными данными №3

Выводы.

Используя полученные знания о рекурсии (в частности, рекурсивных функций), была написана программа на языке программирования C++.

ПРИЛОЖЕНИЕ А

Файл Alg.cpp

```
#include <iostream>
#include <locale.h>

using namespace std;

void fix(int ivector[], int start, int finish){
    if (finish-start < 1){
        return;
    }
    if (finish-start < 2){
        if( ivector[start] > ivector[finish]){
            int c = ivector[start];
            ivector[start] = ivector[finish];
            ivector[finish] = c;
        }
        return;
    }
    fix(ivector, start, start+(finish-start)/2);
    fix(ivector, start+(finish-start)/2+1, finish);
}

int main()
{
    setlocale(0,"");
    int ivector[100];
    int key;
```

```

int icount = 0;

cout << "Введите значения вектора\n";
cin >> key;
while (cin){
    ivector[icount] = key;
    cin >> key;
    icount++;
}

fix(ivector, 0, icount-1);
cout << "{";
for (int i = 0; i < icount-1; i++){
    cout << ivector[i] << ", ";
}
cout << ivector[icount-1] << "}";
return 0;

```