

МИНОБРНАУКИ РОССИИ

Федеральное государственное бюджетное образовательное учреждение высшего
образования



НИЖЕГОРОДСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ
УНИВЕРСИТЕТ им. Р.Е.АЛЕКСЕЕВА

Институт радиоэлектроники и информационных технологий

Кафедра информатики и систем управления

Методы сортировки
(наименование работы)

ОТЧЕТ

по лабораторной работе

по дисциплине

Технологии программирования
(наименование дисциплины)

РУКОВОДИТЕЛЬ:

(подпись)

Капранов С.Н.
(фамилия, и., о.)

СТУДЕНТ:

(подпись)

Шутов А.А.
(фамилия, и., о.)

18-ИСТ-4
(шифр группы)

Работа защищена «__» _____

С оценкой _____

Нижний Новгород, 2020

Содержание

Основная часть отчета	3
Листинг программы	4
Входные и выходные данные.....	6
Используемая литература	7

Основная часть отчета

2 вариант

Задача 1. Имеется N камней веса A_1, A_2, \dots, A_N .

Необходимо разбить их на две кучи таким образом, чтобы веса куч отличались не более чем в 1,5 раза. Если этого сделать нельзя, то указать это.

Листинг программы

(программа написана на языке C++ в среде разработки Visual Studio 2019)

```
#include <iostream>
using namespace std;
int N = 0;
void input(int* a) //проверка на ввод положительного числа
{
    while (!(std::cin >> *a) || *a <= 0)
    {
        std::cin.clear();
        std::cin.ignore(32267, '\n');
        std::cout << "Error" << std::endl;
    }
}
void create_lot(int stones[]) //создание массива весов камней
{
    cout << N << " камней в куче. Их вес:\n";
    for (int i = 0; i < N; i++) {
        cout << (i + 1) << ": ";
        input(&stones[i]);
    }
}
void arrange_lot(int stones[]) //сортировка по убыванию
{
    int i, j, cur;
    for (i = 0; i < N - 1; i++) {
        for (j = i + 1; j < N; j++) {
            if (stones[i] < stones[j]) {
                cur = stones[i];
                stones[i] = stones[j];
                stones[j] = cur;
            }
        }
    }
}
void arrange_heaps(int stones[]) //создание двух куч, как можно более близких по весу
{
    int sum1 = 0, sum2 = 0, k1 = 0, k2 = 0, i;
    int *heap1 = new int[N-1];
    int* heap2 = new int[N-1];
    for (i = 0; i < N; i++) {
        if (sum1 <= sum2) {
            heap1[k1] = stones[i];
            sum1 += stones[i];
            k1++;
        }
        else {
            heap2[k2] = stones[i];
            sum2 += stones[i];
            k2++;
        }
    }
    if (1.0 * sum1 / sum2 <= 1.5) {
        cout << "\nКуча 1: ";
        for (i = 0; i < k1; i++)
            cout << heap1[i] << " ";
        cout << "\nБес 1: " << sum1 << endl;
        cout << "\nКуча 2: ";
        for (i = 0; i < k2; i++)
            cout << heap2[i] << " ";
        cout << "\nБес 2: " << sum2 << endl;
    }
}
```

```
    }  
    else cout << "Вес различается больше чем в 1,5 раза\n";  
    delete[] heap1;  
    delete[] heap2;  
}  
  
int main() {  
    setlocale(LC_ALL, "");  
    cout << "Введите количество камней:" << endl;  
    input(&N);  
    int* lot = new int[N];  
    create_lot(lot);  
    arrange_lot(lot);  
    arrange_heaps(lot);  
    delete[] lot;  
    return 0;  
}
```

Входные и выходные данные

(Рис. 1, 2)

```
Введите количество камней:
6
6 камней в куче. Их вес:
1: 99
2: 5
3: 4
4: 2
5: 36
6: 5
```

```
Введите количество камней:
6
6 камней в куче. Их вес:
1: 3
2: 6
3: 5
4: 4
5: 8
6: 6
```

```
Введите количество камней:
ывав
Error
```

Проверка на ввод положительного числа

Рис. 1. Входные данные

```
Куча 1: 8 5 3
Вес 1: 16

Куча 2: 6 6 4
Вес 2: 16

C:\Users\bons1\source\repos\TP5\Debug\TP5.exe (процесс 13944) завершил работу с кодом 0.
Чтобы автоматически закрывать консоль при остановке отладки, включите параметр "Сервис" ->"Параметры" ->"Отладка" -> "Автоматически закрыть консоль при остановке отладки".
Нажмите любую клавишу, чтобы закрыть это окно...
```

Вес различается больше чем в 1,5 раза

Рис. 2. Выходные данные

Используемая литература

[1] Методические указания по выполнению лабораторных работ (Интернет – ресурс). Режим доступа -

https://vk.com/doc133198571_482602793?hash=ff89b411baad310560&dl=d55fbb202dacc386e9