

Министерство образования и науки РФ
Федеральное государственное бюджетное образовательное учреждение высшего
образования
«Омский государственный технический университет»
Кафедра «Информатика и вычислительная техника»

Отчёт по лабораторной работе № 8
по дисциплине
«Проектирование и тестирование программного обеспечения»

Выполнил:
Студент гр. ПИН-211
Сероухов Е.С. _____
(подп., дата)

Проверил:
Старший преподаватель каф. ИВТ
Карабцов Р.Д. _____
(подп., дата)

ВВЕДЕНИЕ

Цель работы: научиться применять алгоритмы динамического программирования. Решить поставленную задачу.

Задача работы: Задача 8.1 (Веса и меры)

PC/UVaIDs: 111 103/10154

Черепашка по имени Мак очень боится, что его панцирь может сломаться. Поэтому он попросил вас дать совет, как поставить черепашек друг на друга, чтобы построить трон Йертла Черепахи. Каждая из 5607 черепах, призванных Йертлом, обладает своей силой и весом. Ваша задача состоит в том, чтобы составить из них башню максимальной высоты.

Входные данные

Входные данные состоят из нескольких строк, каждая из которых содержит два целых числа, разделенные одним или более пробелом. Эти числа задают вес и силу черепахи. Вес черепахи измеряется в граммах. Сила, которая также измеряется в граммах - это максимальный вес, который способна выдержать черепаха (включая свой собственный). Таким образом, черепаха, весящая 300 грамм и имеющая силу 1000 грамм, может держать 700 грамм на спине. Число черепах не превышает 5607.

Выходные данные

Выходные данные должны состоять из одного целого числа, равного максимальному числу черепах, которые можно поставить стопкой так, чтобы сила ни одной из них не была превышена.

Пример входных данных

300 1000 1000 1200 200 600 100 101

Соответствующие выходные данные

3

РЕЗУЛЬТАТЫ ВЫПОЛНЕНИЯ ЛАБОРАТОРНОЙ РАБОТЫ

1. Для решения задачи был написан код на ЯП Java в среде IntelliJ IDEA 2022.1.2.

Входные данные загружаются из файла “data.txt” (рисунок 1).

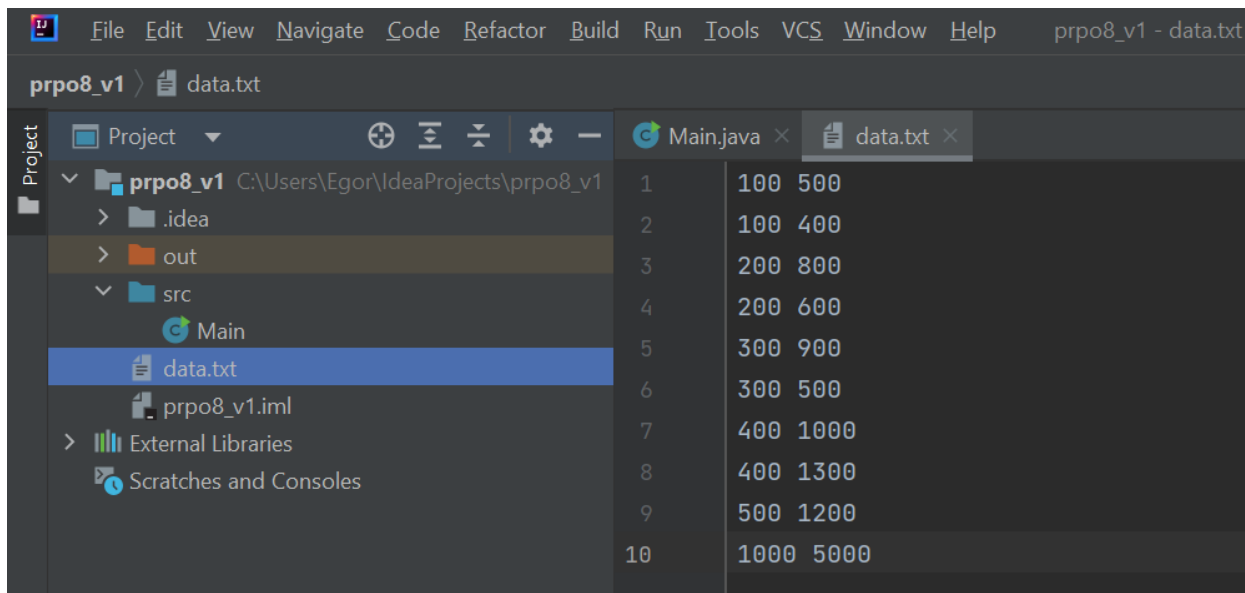


Рисунок 1 – Входные данные

Код программы:

```
import java.io.FileReader;
import java.io.IOException;
import java.util.*;

class Main {
    static class Turtle implements Comparable<Turtle> {
        int w, s;
        public Turtle(int w, int s) {
            this.w = w;
            this.s = s;
        }
        public int compareTo(Turtle other) {
            return Integer.compare(s, other.s);
        }
    }
    static List<Turtle> T = new ArrayList<>();
    static int[] dp = new int[5607];
    public static void solve() {
        Collections.sort(T);

        Arrays.fill(dp, Integer.MAX_VALUE);
        dp[0] = 0;

        int maxNTurtles = 0;

        for (Turtle turtle : T)
            for (int j = maxNTurtles; j >= 0; --j)
                if (turtle.s >= dp[j] + turtle.w && turtle.w + dp[j] < dp[j + 1]) {
                    dp[j + 1] = dp[j] + turtle.w;
                }
    }
}
```

```

        maxNTurtles = Math.max(maxNTurtles, j + 1);
    }

    System.out.println("Максимальное число черепах в
стопке:\n"+maxNTurtles);
}
final static List<String> Data = new ArrayList<>();
public static void dataUploading(String path){
    try (FileReader fr= new FileReader(path)){
        Scanner scan = new Scanner(fr);
        while (scan.hasNextLine())
            Data.add(scan.nextLine());
    }
    catch(IOException ex){
        System.out.println(ex.getMessage());
    }
}

public static void main(String[] args) {
    String file = "data.txt";
    dataUploading(file);
    System.out.println("Входные данные загружены из файла: " + file);
    System.out.println(" Вес | Сила");
    for (String datum : Data){
        int[] numArr = Arrays.stream(datum.split("
")).mapToInt(Integer::parseInt).toArray();
        int w = numArr[0];
        int s = numArr[1];
        System.out.printf("%4d | %4d\n",w,s);
        T.add(new Turtle(w, s));
    }

    solve();
}
}

```

2. Было проведено тестирование работы программы (рисунок 2).

The screenshot shows an IDE window with the following content:

- Run:** C:\Users\Egor\.jdk\openjdk-19\bin\java.exe "-javaage...
- Output:**

```

Входные данные загружены из файла: data.txt
Вес | Сила
100 | 500
100 | 400
200 | 800
200 | 600
300 | 900
300 | 500
400 | 1000
400 | 1300
500 | 1200
1000 | 5000
Максимальное число черепах в стопке:
7
Process finished with exit code 0

```

Рисунок 2 – Тестирование работы программы.

ЗАКЛЮЧЕНИЕ

Вывод: в ходе работы была написана программа на ЯП Java. Программа решила поставленную задачу и прошла проверку при тестировании.