

## ПЛАНЕТИ

Планетите в една планетна система се характеризират с:

- име (непразен низ, евентуално съдържащ интервали);
- номер – по отдалеченост от централното светило;
- брой спътници – цяло неотрицателно число;
- маса (в земни маси – положително число, не непременно цяло).

Първият ред на текстовия файл **PLANETS.IN** съдържа естественото число  $N$ , определящо броя на планетите. Всеки от останалите  $2N$  реда съдържа последователно:

- ако е с четен номер – низ, определящ името планета от системата;
- ако е с нечетен номер – три числа, разделени с интервал, първото от които определя номера на планетата от предишния ред, второто – броя луни, а третото – масата спрямо Земята.

Задача 1: Осигурете въвеждане на данните от файла.

Задача 2: Изведете на стандартния изход номериран списък на имената планетите при следния приоритет на свойствата: най-значимо – маса, след това дължина на името и накрая отдалеченост от слънцето. Номерът в списъка е в 3 символа (с евентуално водещи интервали), завършва с точка и се отделя с интервал от името.

Задача 3. От стандартния вход се въвежда низ, не по-дълъг от 10 символа. Намерете планетите (ако има такива), в името на които се съдържа този низ (без да се различават главни и малки букви). Изведете списъка от зад. 2 за тях, като по-напред поставите тези, които имат по-голям брой спътници.

ПРИЕМРЕН ВХОДЕН ФАЙЛ:

```
4
nova zemya
2 3 1
venerka
4 10 0.8
jupitercho
5 0 1803.5
america v nebeto
6 1 1
```

ИЗХОД ЗА ЗАД. 2:

1. jupitercho
2. america v nebeto
3. nova zemya
4. venerka

Ако за зад. 3 е въведен низът „er”, то

ИЗХОД ЗА ЗАД. 3:

1. venerka
2. america v nebeto
3. jupitercho

# Реализация

## Клас Planet

```
public class Planet{
    //Свойства
    private String name; //име
    private int number,sat; //номер и брой спътници
    private double mass; //маса
    //Конструктор по всички данни
    public Planet(String nm,int n,int s,double m){
        name=nm;
        number=n;
        sat=s;
        mass=m;
    }
    //Гетъри
    public String getName(){
        return name;
    }
    public int getNumber(){
        return number;
    }
    public double getMass(){
        return mass;
    }
    public int getSat(){
        return sat;
    }
}
```

## Клас Cmp1

```
import java.util.Comparator;
//Клас за подредба по приоритетите на зад 2
public class Cmp1 implements Comparator<Planet> {
    public int compare(Planet a,Planet b){
        if (a.getMass(>b.getMass()) return -1;//По-голяма маса
        if (a.getMass(<b.getMass()) return 1; //По-малка маса
        int t=a.getName().length()-b.getName().length();//Сравняване на дължините на името
        if (t!=0) return -t; //Ако не са еднакво дълги - обратно на дължината
        return b.getNumber()-a.getNumber(); //Обратно на близостта до светилото
    }
}
```

## Клас Cmp2

```
import java.util.Comparator;
//Клас за намаляваща подредба по брой сателити
public class Cmp2 implements Comparator<Planet> {
    public int compare(Planet a,Planet b){
        return b.getSat()-a.getSat();
    }
}
```

## Клас Main

```
import java.util.Scanner;
import java.io.File;
import java.util.ArrayList;
public class Main {
    //Метод за извеждане на номериран списък от динамичен масив L
    // от планети, в името на които се съдържа низ s
    private static void show(ArrayList<Planet> a,String s){
        for (int i=0,j=1;i<a.size();i++) if (a.get(i).getName().indexOf(s)>=0) {
            if (i<10) System.out.print(' ');
            if (i<100) System.out.print(' ');
            System.out.println(j+++" "+a.get(i).getName());
        }
    }
    public static void main(String[] args) {
        File file; //Входен файл
        ArrayList<Planet> L=new ArrayList<Planet>();//Динамичен списък с планети
        Scanner inp;
        //Променливи за четене на параметрите от файла
```

```

int n, no, sat;
String s;
double m;
try{ //Опит за отваряне на входния файл
    file=new File("PLANETS.in");
    inp=new Scanner(file);
}catch(Exception e){
    System.out.println("File not found"); //Съобщение при неуспех
    return;
}
n=inp.nextInt();//Четене на броя описани във файла планети
inp.nextLine(); //Изчистване на входния буфер
for(int i=0;i<n;i++){
    s=inp.nextLine(); //Име на планетата
    no=inp.nextInt(); //Номер,
    sat=inp.nextInt();// брой сателити,
    m=inp.nextDouble();// маса
    inp.nextLine();//Изчистване на входния буфер
    L.add(new Planet(s, no, sat, m)); //Добавяне към масива на нова планета
        //с прочетените свойства
}
inp.close();//Затваряне на входния файл
L.sort(new Cmp1()); //Сортиране по приоритетите на зад. 2
show(L, ""); //Извеждане на номерирания списък (празният низ е подниз на всеки низ)
inp=new Scanner(System.in); //За четене от стандартния вход
System.out.print("Planet name should contain: "); //Подсказка
s=inp.nextLine();//Търсен подниз
inp.close();
L.sort(new Cmp2()); //Сортиране намаляващо по брой сателити
show(L, s); //Извеждане на номерирания списък
}
}

```