Министерство цифрового развития, связи и массовых коммуникаций Российской Федерации Ордена Трудового Красного Знамени

Федеральное государственное бюджетное образовательное учреждение высшего образования

«Московский технический университет связи и информатики»

Отчет по задачам №1

на тему: «Методы сортировки»

Выполнил: студент группы БВТ1903

Белов Сергей Павлович

Проверил:

Павликов Артём Евгеньевич

# **Драфт НБА**

Задача: Перед каждым сезоном НБА проходит драфт, то есть церемония выбора игроков командами. Перед драфтом скаутские службы команд детально изучают потенциальных новичков. Одна из команд поручила вам написать программу для обработки собранных данных в соответствии с внутрикомандными стандартами.

Каждый из новичков до этого провёл как минимум один сезон в студенческих лигах, так что для каждого игрока известны пять основных целочисленных параметров:

● Рост игрока – ожидаемый диапазон от 190 до 220 см.

● Размах рук (иначе говоря, wingspan) – ожидаемый диапазон от 200 до 250 см.

● Среднее количество очков за матч – ожидаемый диапазон от 10 до 20.

● Среднее количество подборов за матч – ожидаемый диапазон от 2 до 6.

● Среднее количество передач за матч – ожидаемый диапазон от 3 до 7.

Таким образом, каждый из параметров может оказаться ниже ожидаемого диапазона, в нижней половине ожидаемого диапазона, в верхней половине ожидаемого диапазона (середина входит в верхнюю половину) и выше ожидаемого диапазона.

Перед драфтом требуется распределить игроков по следующим категориям:

● уникальный игрок (таких ещё называют «единорогами» — unicorn) с сочетанием выдающихся физических данных и игровых навыков;

● игрок, достойный выбора в первом раунде;

● игрок, достойный выбора во втором раунде;

● игрок, которого не стоит выбирать вообще.

Если у игрока три параметра выше ожидаемого диапазона, причём среди них обязательно есть рост или размах рук, то игрок считается «единорогом» (категория 0)

Игрока рекомендуется выбирать в первом раунде драфта (категория 1), если верно одно из следующих утверждений:

● У игрока два параметра выше ожидаемого диапазона и ещё один – как минимум в верхней половине ожидаемого диапазона.

● У игрока все параметры как минимум в ожидаемом диапазоне и не менее трёх – как минимум в верхней половине.

Игрока рекомендуется выбирать во втором раунде драфта (категория 2), если верно одно из следующих утверждений:

● У игрока один параметр выше ожидаемого диапазона и ещё один — как минимум в верхней половине ожидаемого диапазона.

● У игрока три параметра как минимум в верхней половине ожидаемого диапазона.

В остальных случаях тратить выбор драфта на этого игрока не рекомендуется (категория 3).

Вам заданы параметры некоторых из 32 игроков текущего драфта (для объективности имена игроков не указаны). Ваша задача – для каждого игрока определить категорию, под которую он подпадает для вашей команды на драфте.

Алгоритм реализации:

Для реализации поставленной задачи было решено создать систему очков, которые будут добавляться каждому потенциальному игроку при наличии какой-либо характеристики. По сумме этих очков будет определяться категория игрока. Очки начисляются следующим образом:

* Если характеристика выше ожидаемого диапазона, то начисляется 1000 очков
* Если характеристика находится в верхней половине ожидаемого диапазона, то начисляется 100 очков
* В противном случае очки не начисляются

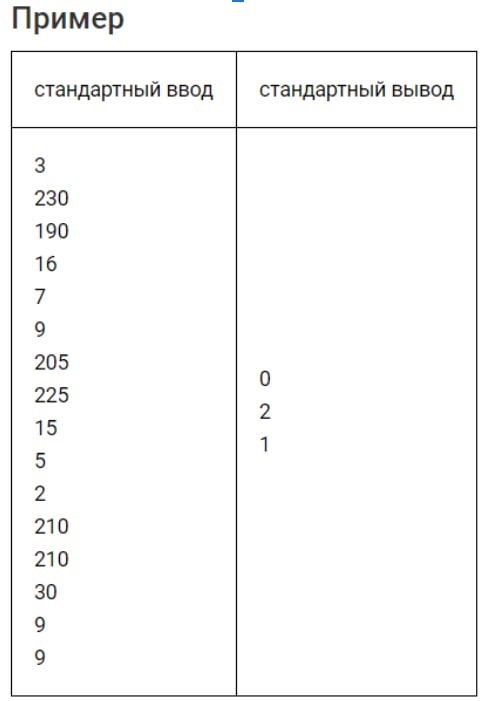
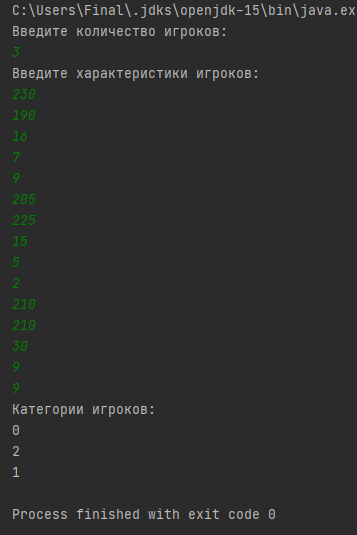
Для определения группы по сумме очков установлены следующие условия:

* Если набрано не менее 3000 (то есть три параметра выше ожидаемого) очков, то дополнительно проверяется. Что среди категорий, за которые игрок получил 1000 очков. Есть Руки и/или Рост, если этого не происходит, то категория понижается до первой
* Если набрано 2100 очков (то есть два параметра выше диапазона и один в верхней половине) игрок попадает в первую категорию
* Если набрано 1100 очков (то есть 1 параметр выше диапазона и 2 в верхней половине) или 300 очков (три параметра в верхней половине диапазона). То игрок попадает во вторую категорию
* В противном случае категория не определяется

Листинг программы:

import java.io.Console;  
import java.util.Scanner;  
  
public class Basketball {  
  
 public static void main (String[] arg){  
  
 System.*out*.println("Введите количество игроков:");  
 int number;  
 Scanner s = new Scanner(System.*in*);  
 number = s.nextInt();  
 int players[][] = new int[number][5];//players\*characteristics  
  
 System.*out*.println("Введите характеристики игроков:");  
 for (int i=0;i<number;i++){  
 for (int j=0;j<5;j++){  
 players[i][j]=s.nextInt();//get characteristics all players  
 }  
 }  
  
 System.*out*.println("Категории игроков:");  
 for (int i=0;i<number;i++){  
 int Counter=0;  
 //get summary of all characteristics  
 Counter+=*Height*(players[i][0]);  
 Counter+=*Arms*(players[i][1]);  
 Counter+=*Points*(players[i][2]);  
 Counter+=*Take*(players[i][3]);  
 Counter+=*Pass*(players[i][4]);  
  
 if (*Categories*(Counter)==0){//unicorn?  
 if (*Arms*(players[i][1])==1000||*Height*(players[i][0])==1000){//3000 points and arm or height characteristic?  
 System.*out*.println(*Categories*(Counter));//Unicorn!  
 }  
 else System.*out*.println(*Categories*(Counter)+1);//we have first category because height or arms not unicorn  
 }  
 else System.*out*.println(*Categories*(Counter));//not unicorn-> print category   
 }  
 }  
  
 public static int Height (int height){  
 //input: player characteristic Height  
 int counts=0;  
 if (height>220){ //extra  
 counts=1000;  
 }  
 else if (height<=220&&height>=205){ //upper range  
 counts=100;  
 }  
 else counts=0;  
 return counts;  
 }  
  
 public static int Arms (int arm){  
 //input: player characteristic Arms  
 int counts=0;  
 if (arm>250){ //extra  
 counts=1000;  
 }  
 else if (arm<=250&&arm>=225){ //upper range  
 counts=100;  
 }  
 else counts=0;  
 return counts;  
 }  
  
 public static int Points (int points){  
 //input: player characteristic Points  
 int counts=0;  
 if (points>20){ //extra  
 counts=1000;  
 }  
 else if (points<=20&&points>=15){ //upper range  
 counts=100;  
 }  
 else counts=0;  
 return counts;  
 }  
  
 public static int Take (int take){  
 //input: player characteristic Take  
 int counts=0;  
 if (take>6){ //extra  
 counts=1000;  
 }  
 else if (take<=6&&take>=4){ //upper range  
 counts=100;  
 }  
 else counts=0;  
 return counts;  
 }  
  
 public static int Pass (int pass){  
 //input: player characteristic Pass  
 int counts=0;  
 if (pass>7){ //extra  
 counts=1000;  
 }  
 else if (pass<=7&&pass>=5){ //upper range  
 counts=100;  
 }  
 else counts=0;  
 return counts;  
 }  
  
  
 public static int Categories (int points){  
 //input: summary of player's counts  
 if (points>=3000){ //may be the unicorn  
 return 0;  
 }  
 else if (points>=2100){ //2 \*1000 (extra)+100 (upper range)  
 return 1;  
 }  
 else if (points>=1100||points>=300){//1000 (extra)+100 (upper range) or 3\*100(upper range)  
 return 2;  
 }  
 else return 3;  
  
 }  
}

Результат работы программы:

Исходные данные: Полученные результаты: 

# **Вывод**

В результате выполнения задания был разработан и реализован свой алгоритм драфта для баскетбольной лиги по категориям игроков.