**Nivel 1 Problema 1 truco Certamen Nacional 2009**

**Tirando reyes**

**Descripción del problema**

Al jugar un partido de truco entre amigos, se deben armar dos equipos. Según la tradición, para armar los equipos se "tiran reyes" antes de empezar a jugar.

Este procedimiento consiste en repartir las cartas de una baraja española de **40** cartas, cara hacia arriba para que todos las vean, de a una y en el orden de la ronda hasta que salga un rey (carta con el número **12** de cualquier palo).

Cuando un jugador recibe un rey sale de la ronda y no recibe más cartas. Luego se sigue repartiendo las siguientes continuando la ronda donde se encontraba.

Cuando **la mitad** de los jugadores hayan recibido una carta con un rey, los dos equipos quedan conformados como los que recibieron un rey y los que no.

Los **n** amigos están numerados de **1** a **n** y comienza la ronda entregando la primer carta del mazo al jugador **1**, luego al **2** y así hasta completar la ronda.

Para poder controlar si los jugadores sortean bien los equipos deberás escribir un programa truco.cpp, truco.c o truco.pas que simule en computadora el sorteo.

**Datos de entrada:**

Se recibe un archivo truco.in del directorio actual, que contiene:

Primera línea: La cantidad **n** de amigos que se aprestan a jugar ( **n** = **4, 6** u **8** ).

**40** líneas adicionales, cada una de las cuales describe una carta del mazo con un número separado por un espacio de una letra. El número está en el rango **1..7** o **10..12**. La letra puede ser **O**, **C**, **E** o **B**.

**Datos de salida:**

El programa debe generar el archivo truco.out, en el directorio actual con dos líneas:

La primera línea debe tener el equipo conformado por quienes recibieron los reyes, representados por el número de jugador y en el orden en que recibieron los reyes, separados por un espacio.

La segunda línea debe tener el equipo conformado por los que no recibieron los reyes, en el orden creciente de la ronda, separados por un espacio.

**Ejemplo :**

Si el archivo truco.in contiene:

6

3 C

5 B

1 C

4 C

12 E

6 C

12 B

10 O

10 E

2 O

7 B

4 E

7 O

1 B

4 O

1 O

4 B

11 E

6 O

2 C

10 C

5 O

11 O

3 O

6 E

12 C

12 O

7 C

3 E

2 E

5 E

1 E

3 B

7 E

10 B

11 B

11 C

6 B

5 C

2 B

El archivo truco.out debe contener

5 1 4

2 3 6

import java.io.BufferedReader;

import java.io.File;

import java.io.FileReader;

import java.io.FileWriter;

import java.io.IOException;

import java.io.PrintWriter;

import java.util.Iterator;

public class Archivo {

public String rutaEntrada;

public String rutaSalida;

public Archivo(String entrada, String salida) {

this.rutaEntrada = entrada;

this.rutaSalida = salida;

}

public void leerArchivo(Reyes reyes) {

File file;

FileReader fr = null;

BufferedReader br = null;

String[] datos;

try {

file = new File(this.rutaEntrada);

fr = new FileReader(file);

br = new BufferedReader(fr);

reyes.cantJugadores = Integer.parseInt(br.readLine());

for(int i=0; i<40; i++){

Carta carta = new Carta();

datos = br.readLine().split(" ");

carta.letra = datos[1];

carta.numero = Integer.parseInt(datos[0]);

reyes.cartas.addElement(carta);

}

} catch (IOException e) {

e.printStackTrace();

} catch (Exception e1) {

e1.printStackTrace();

} finally {

try {

if (fr != null)

fr.close();

} catch (Exception e2) {

e2.printStackTrace();

}

}

}

public void grabarArchivo(Reyes reyes) {

FileWriter fw = null;

PrintWriter pw = null;

try {

fw = new FileWriter(this.rutaSalida);

pw = new PrintWriter(fw);

Iterator<Integer> ite = reyes.equipo1.iterator();

while(ite.hasNext()){

pw.print(ite.next());

pw.print(" ");

}

pw.println();

ite = reyes.equipo2.iterator();

while(ite.hasNext()){

pw.print(ite.next());

pw.print(" ");

}

} catch (IOException e) {

e.printStackTrace();

} catch (Exception e1) {

e1.printStackTrace();

} finally {

try {

if (fw != null)

fw.close();

} catch (Exception e2) {

e2.printStackTrace();

}

}

}

}

public class Carta {

public int numero;

public String letra;

}

public class Test {

public static void main(String[] args) {

Archivo arch;

Reyes reyes;

arch = new Archivo("truco.in", "truco.out");

reyes = new Reyes();

arch.leerArchivo(reyes);

reyes.tirandoReyes();

arch.grabarArchivo(reyes);

}

}

import java.util.Iterator;

import java.util.Vector;

public class Reyes {

public int cantJugadores;

public Vector<Carta> cartas;

public Vector<Integer> equipo1;

public Vector<Integer> equipo2;

public Reyes(){

cartas = new Vector<Carta>();

this.equipo1 = new Vector<Integer>();

this.equipo2 = new Vector<Integer>();

}

public void tirandoReyes() {

int nroCarta = 0;

while ((cantJugadores/2)> this.equipo1.size())

for (int i = 0; i < this.cantJugadores && nroCarta<40

&& (cantJugadores/2)> this.equipo1.size(); i++) {

if (!this.fueElegido(i+1)) {

if (cartas.get(nroCarta).numero == 12) {

this.equipo1.addElement(new Integer(i + 1));

}

nroCarta++;

}

}

for (int i = 1; i <= this.cantJugadores; i++) {

if (!this.fueElegido(i)) {

this.equipo2.addElement(new Integer(i));

}

}

}

private boolean fueElegido(int nro) {

Iterator ite = (Iterator) this.equipo1.iterator();

while (ite.hasNext()) {

Integer jugador = (Integer) ite.next();

if (jugador.intValue() == nro)

return true;

}

return false;

}

}