

Práctica Lotería primitiva

Se pretende desarrollar una aplicación que permita rellenar boletos de lotería primitiva con unas funcionalidades similares a las de la práctica Quiniela. Cada jugador podrá hacer un máximo de 20 primitivas por jornada. Dado que este límite está sujeto a las políticas de lucha contra la ludopatía puede sufrir cambios más adelante por lo que se guardará en una variable *final* llamada "LIMITE_PRIMITIVAS".

Todos los boletos rellenados por un jugador se almacenarán en un array creado para la ocasión. En esa estructura NO estará incluido el resultado oficial de ese sorteo.

En el menú a desarrollar, se pueden observar las distintas opciones que se mostrarán al usuario cuando se ejecute la aplicación, es el que se muestra en la siguiente captura de pantalla:

```

                Menú lotería primitiva
=====
1. Rellenar un boleto de lotería primitiva nuevo manualmente
2. Rellenar un boleto de lotería primitiva nuevo aleatoriamente
3. Seleccionar el boleto de lotería primitiva activo
4. Mostrar el boleto de lotería primitiva activo
5. Comprobar los aciertos del boleto de lotería primitiva activo
6. Comprobar los aciertos de todos los boletos jugados
7. Mostrar la combinación ganadora
8. Salir de la aplicación

Elige la opción deseada (1-8):
```

Este menú deberá mostrarse de nuevo al usuario, tras atender cada petición, hasta que se pulse la opción de salir de la aplicación.

La opción 1 irá pidiendo, uno a uno, seis números entre el 1 y el 49 para cada una de las seis casillas que debe completar. Evidentemente hay que comprobar que los valores introducidos están en ese rango y emitir un mensaje de error en caso contrario. Se indicará qué número de casilla se está cumplimentando.

Programación

Elige la opción deseada (1-8): 1

Opción elegida: 1

Introduzca un número del 1 al 49 para la casilla: 1 : 5

Recibida casilla 1

Introduzca un número del 1 al 49 para la casilla: 2 : 18

Recibida casilla 2

Introduzca un número del 1 al 49 para la casilla: 3 : 85

Error, los únicos números posibles son del 1 al 49, por favor, inténtelo de nuevo

Introduzca un número del 1 al 49 para la casilla: 3 :

Otra de las situaciones que se debe vigilar es que no se repitan números previamente introducidos para ese boleto, si esto fuera detectado se indicará:

Introduzca un número del 1 al 49 para la casilla: 3 : 17

Recibida casilla 3

Introduzca un número del 1 al 49 para la casilla: 4 : 17

Error, el número 17 ya está en la combinación, no se pueden repetir.

Introduzca un número del 1 al 49 para la casilla: 4 :

Las opciones 1 y 2 permiten generar boletos manual o aleatoriamente, en ambos casos se mostrará un mensaje indicando el éxito de la ejecución. El reintegro se generará aleatoriamente siempre, aunque se rellene manualmente la combinación jugada.

Elige la opción deseada (1-8): 2

Opción elegida: 2

Boleto rellenado correctamente

Cada vez que se rellene un boleto nuevo éste se agregará a los boletos que haya realizado el jugador en ese sorteo. Ambas opciones, manual y aleatoria, estarán sujetas al límite máximo establecido por sorteo y jugador. Si se tratará de vulnerar ese límite habrá un mensaje advirtiendo de que ya se ha alcanzado el tope.

Programación

```
Elige la opción deseada (1-8): 2
Opción elegida: 2
Límite de primitivas alcanzado. Juega con responsabilidad
```

La opción 3 permite seleccionar cuál de los boletos jugados es el activo por si posteriormente se quiere mostrar o comprobar sus aciertos. No hay ningún boleto activo por defecto y si se elige uno incorrecto se hará activo el boleto número 1. Estas circunstancias serán notificadas mediante mensajes por pantalla.

```
Elige la opción deseada (1-8): 3
Opción elegida: 3
No has rellenado ningún boleto aún

Elige la opción deseada (1-8): 3
Opción elegida: 3
Elige el boleto activo (1-1): 5
Boleto elegido: 5
Boleto elegido incorrecto. Se activa el boleto 1
```

Por medio de la cuarta opción se mostrará el contenido del boleto activo en ese momento:

```
Elige la opción deseada (1-8): 4
Opción elegida: 4
Números: 6   14   19   23   38   43
Reintegro: 2
```

La quinta opción contrasta el boleto activo con los resultados oficiales del sorteo para comprobar cuantos aciertos tiene ese boleto:

Programación

Elige la opción deseada (1-8): 5

Opción elegida: 5

Aciertos: 0 + Complementario

También se pueden comprobar en una sola operación los aciertos de todos los boletos jugados en el mismo sorteo por un jugador, se hace con la opción seis:

Elige la opción deseada (1-8): 6

Opción elegida: 6

Se muestran los resultados de todos los boletos jugados

Boleto 1

Aciertos: 0 + Complementario

Boleto 2

Aciertos: 3

.

.

.

Boleto 17

Aciertos: 0

Boleto 18

Aciertos: 1 Reintegro acertado

Boleto 19

Aciertos: 1

Boleto 20

Aciertos: 1

La opción siete muestra la combinación ganadora:

```
Elige la opción deseada (1-8): 7
Opción elegida: 7

      COMBINACIÓN GANADORA
      =====
Números: 1   5   7   13  17  36
Complementario: 38      Reintegro: 1
```

Por último, la opción ocho sale de la aplicación mostrando un mensaje de despedida:

```
Elige la opción deseada (1-8): 8
Opción elegida: 8

Gracias por usar Lotería Primitiva
Aplicación terminada
```

Consideraciones técnicas

1. Desarrolla, al menos, dos clases para realizar esta práctica: la principal y *BoletoPrimitiva*.
2. Un boleto de lotería primitiva consta de una combinación de seis números elegibles del 1 al 49 más un reintegro asignado aleatoriamente del 0 al 9. Los boletos oficiales también incorporan un número complementario del 1 al 49, distinto de los seis que conforman la combinación ganadora.
3. Implementa la clase *BoletoPrimitiva* usando para su atributo *combinacionJugada* un *TreeSet*.
4. Codifica los métodos que consideres necesarios para conseguir las funcionalidades pedidas en este enunciado.
5. Solamente los boletos oficiales (los que albergan la combinación ganadora resultante del sorteo oficial) tienen en el atributo *complementario* un valor válido, los boletos que instancia cada jugador no.
6. El boleto oficial del sorteo no se almacenará en la misma estructura de los boletos del jugador, como ya se ha indicado en la página 1 de este documento.

Programación

Cualquier implementación que no respete estas consideraciones será calificada como 0.