

Actividad 5.14 – Generador de apuestas de lotería

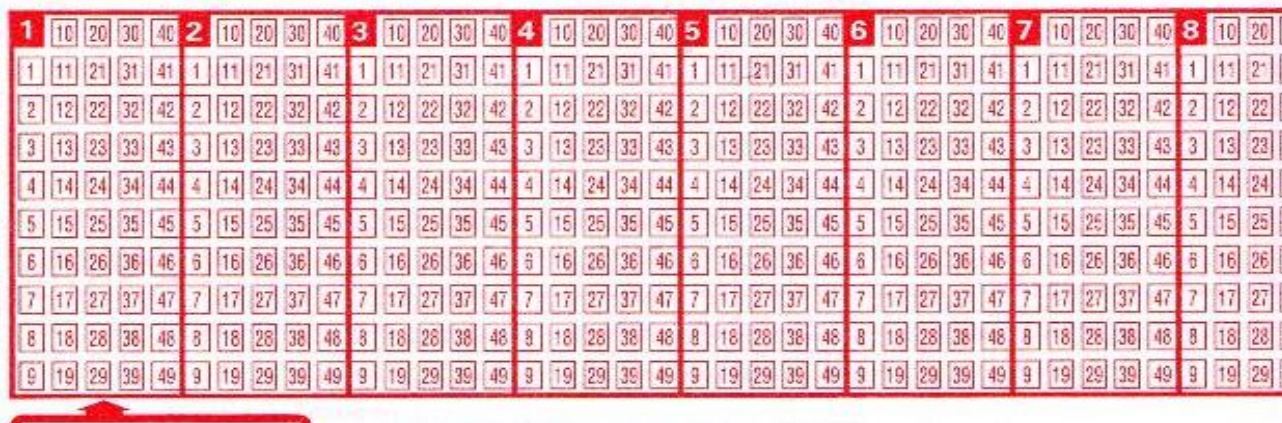
Enunciado

Vamos a implementar un programa que nos permita generar boletos de la lotería primitiva. Un boleto puede tener hasta **8 apuestas** y **cada apuesta** está compuesta por **6 números aleatorios** (del 1 al 49) que no **pueden repetirse** (en la misma apuesta).

En el programa, los números de la apuesta **aparecen siempre ordenados** de menor a mayor.

El programa debe ser capaz de:

- Preguntar al usuario cuántas apuestas quiere realizar en el boleto
- Mostrar por pantalla las apuestas junto al coste de éstas, atendiendo a la tabla de precios indicada más abajo.
- Simular y mostrar un sorteo. Un sorteo está compuesto por una lista de 6 números aleatorios, más un número adicional que será el complementario.
- Determinar el número de aciertos de cada apuesta generada.
- Determinar el premio que cada apuesta consigue, atendiendo la tabla de premios indicada más abajo.
- Determinar el precio que el usuario paga por el número de apuestas determinadas.



BOLETO DE LOTERÍA PRIMITIVA

TABLA DE PRECIOS

N.º apuestas	Precio
1	1,25€
2 - 3	1,5€ / apuesta
4 - 5	2€ / apuesta
6 - 7	12€ / boleto
8	18€ / boleto

TABLA DE PREMIOS SEGÚN ACIERTOS

Nivel de acierto	Aciertos	Premio (€)
1	6	500000
2	5 + complementario	50000
3	5	10000
4	4	500
5	3	50
6	2	0
7	1	0
8	0	0

NORMAS DE IMPLEMENTACIÓN

- Debes utilizar métodos para cada una de las funcionalidades requeridas y mantener el método `main()` lo más simple posible.
- Para facilitar la resolución de la actividad, se indican más abajo las cabeceras de algunos métodos que deberás **implementar de manera obligatoria** en tu programa.
- Para ayudar a la comprensión del problema, **se presenta un ejemplo de ejecución** del programa. Recuerda mantener el formato que se te presenta.
- Ten en cuenta que **no está permitida la utilización de variables de ámbito global**, excepto el objeto Scanner. De otro modo, se dará la práctica por no evaluada.
- Se pueden hacer uso de **constantes**.
- Si se introduce un número de apuestas incorrectas (un número fuera de rango o algo que no sea un número) se mostrará un texto de error, volviendo a solicitar el dato.

Ejemplo de ejecución:

```
¿Cuántas apuestas quieres realizar?: hola
Apuestas incorrectas. Introdúcelas de nuevo.

¿Cuántas apuestas quieres realizar?: 10
Apuestas incorrectas. Introdúcelas de nuevo.

¿Cuántas apuestas quieres realizar?: 4

----- Apuesta 1 -----
+---+---+---+---+---+---+
| 02 | 04 | 10 | 15 | 43 | 49 |
+---+---+---+---+---+---+

----- Apuesta 2 -----
+---+---+---+---+---+---+
| 14 | 20 | 24 | 35 | 42 | 49 |
+---+---+---+---+---+---+

----- Apuesta 3 -----
+---+---+---+---+---+---+
| 02 | 08 | 14 | 22 | 23 | 32 |
+---+---+---+---+---+---+
```

----- Apuesta 4 ----

```
+---+---+---+---+---+---+
| 14 | 15 | 28 | 41 | 45 | 49 |
+---+---+---+---+---+---+
```

---- Total a Pagar ----

```
+---+---+---+---+---+---+
| Apuestas: |    4    |
+---+---+---+---+---+---+
| A Pagar   |  08,00€  |
+---+---+---+---+---+---+
```

----- SIMULADOR DE SORTEO -----

```
+---+---+---+---+---+---+---C---+
| 14 | 28 | 32 | 41 | 45 | 49 | 15 |
+---+---+---+---+---+---+---+
```

----- PREMIOS OBTENIDOS -----

Apuesta 1: 1 acierto (0 €)
Apuesta 2: 2 aciertos (0 €)
Apuesta 3: 2 aciertos (0 €)
Apuesta 4: 5 aciertos + complementario (50000 €)

Total ganado: 50000€



Recuerda que para mostrar las apuestas, la función **printf** te permite mostrar un número con un **número constante de dígitos**, de forma que si el número no los tiene, lo completará con el carácter especificado (**padding**)

```
System.out.printf("%010d\n", 40); //mostrará el dígito 40 con 10
                                     dígitos '0' por la izquierda
```



Para facilitar la resolución del programa, recuerda que debes hacer métodos que hagan una función única. Como mínimo deberás tener estos métodos:

```
/**
 * Genera una apuesta de 6 números
 */
public static int[] generarApuesta()

/**
 * Muestra la @combinación recibida en el formato de salida
```

```
* especificado en el enunciado  
**/
```

```
public static void mostrarCombinacion(int[] combinacion)
```

```
/**  
 * Genera un sorteo de 7 números (6 más el complementario)  
 **/
```

```
public static int[] generarSorteo()
```

```
/**  
 * Obtiene el nivel de acierto de una @apuesta en función de la  
 * @combinacionGanadora (ver tabla del enunciado)  
 **/
```

```
public static int obtenerNivelDeAcierto(int[] apuesta, int[] combinacionGanadora)
```

```
/**  
 * Muestra el número de aciertos y el premio obtenido de cada apuesta (se asume *  
 * que @nivelesDeAcierto están dispuestos según el orden de cada apuesta  
 * generada, es decir, el primer nivel de premio corresponde a la apuesta 1, y  
 * así sucesivamente  
 **/
```

```
public static void mostrarPremiosObtenidos(int[] nivelesDeAcierto)
```

```
/**  
 * Obtiene el total del premio a partir de los premios de cada apuesta realizada  
 * guardados en @nivelesDeAcierto  
 **/
```

```
public static int obtenerTotalPremio(int[] nivelesDeAcierto)
```

```
/**  
 * Muestra una tabla donde se muestra el @numApuestas y el total a pagar por  
 * el boleto según ese dato (la tabla Total a pagar del ejemplo). Bájate en los  
 * precios de la tabla del enunciado.  
 **/
```

```
public static void mostrarTotalAPagar(int numApuestas)
```