

Asignatura: “Estructuras de Datos”

Práctica Obligatoria de la Asignatura: “Simulación del juego del bingo”

Normas

- La práctica es *obligatoria* y tiene un *peso del 30%* sobre la nota final.
- Se realizará en grupos de tres alumnos. Los grupos deberán apuntarse a través de una actividad llamada “Elección de Grupos para la Práctica Obligatoria” que se ha creado en el Aula Virtual (en la pestaña “Evaluación”).
- Como parte de la evaluación esta práctica, se prevé la realización de una presentación de la misma al profesor de la asignatura (más adelante se indicará el procedimiento).
- Cada grupo deberá entregar su solución a la práctica (solo una por grupo) en un archivo zip a través del Aula Virtual de la asignatura. El archivo comprimido con la entrega contendrá: 1) el código fuente de la solución planteada a la práctica; 2) una breve memoria explicativa describiendo su funcionamiento y los desarrollos realizados; y 3) un archivo “jar” llamado “practica.jar” que permita ejecutar directamente desde consola la solución desarrollada.
- La *fecha límite* de entrega de la práctica es el *18 de mayo de 2022* (día del examen final).

Objetivo de la práctica

Se trata de diseñar e implementar en Java una aplicación que simule el juego del bingo. El objetivo principal de esta práctica es seleccionar y codificar adecuadamente los Tipos Abstractos de Datos (TADs) que intervienen en el diseño de la solución al problema. A continuación, se describen las reglas básicas del “juego del bingo” (en Internet pueden encontrarse muchas páginas explicativas sobre dicho juego, incluyendo su historia). Después, se plantean unas reglas específicas, que habrá que tenerlas en cuenta obligatoriamente en el desarrollo de la solución a esta práctica. Finalmente, se mostrará mediante un ejemplo concreto la salida por pantalla correspondiente a una posible partida de bingo.

Descripción general del juego.

El *bingo* es un juego de azar que consta de: un bombo, un total de **90 bolas numeradas** en su interior, y unos **cartones con números impresos** que se reparten entre los **jugadores**. Al comienzo de cada partida, cada uno de los jugadores participantes recibe un cartón que contiene **15 números aleatorios diferentes**, dentro del rango entre 1 y el 90. Los números se distribuyen en **3 líneas**, cada una de las cuales contiene **5 números impresos** y **4 casillas sin número** (o huecos). Dentro de cada columna del cartón pueden aparecer **entre uno o dos números**; en el caso de aparecer dos números, éstos **aparecen ordenados** de forma creciente en cada columna (ver Fig. 1).

en cada fila

Un **locutor** diciendo consecutivamente números de forma aleatoria y va sacando las bolas del bombo correspondientes dichos números. Si un jugador (o varios de ellos) tiene(n) dicho número, lo **tacha(n)** de su cartón, y el juego continúa así hasta que alguien consigue marcar todos los números de su cartón (premio del “bingo”), convirtiéndose en el ganador de la

partida. Existe también un premio adicional, que se consigue antes de acabar una partida, llamado “línea”, para el **primer jugador que tache todos los números de una línea** en su cartón asignado. La siguiente figura muestra un ejemplo de un posible cartón para jugar al bingo.

Cartón n.º 1

5	10			44		62	70	
	16		37	47			76	81
7		21	39		58			89

www.cartones-bingo.net

Fig. 1. Ejemplo de cartón para el juego del bingo.

Simulación del juego del bingo: estructuras de datos y reglas del juego.

Se pide **diseñar e implementar en Java** una aplicación para simular partidas de bingo entre un número variable de jugadores humanos, comprendido **entre 2 y 4 jugadores**. El número de jugadores se introducirá al comienzo de la partida; si no se indica ningún número, por defecto jugarán 2 personas. El ordenador se encargará de controlar toda la “logística” del juego: es decir, generar aleatoriamente la secuencia de números sacados del bombo y llevar el control de números aparecidos; leer los cartones de cada jugador, que aparecen inicialmente guardados en un archivo de texto (consultar en Internet la estructura de un cartón de bingo válido); “tachar” en cada jugada el número “extraído del bombo” de los cartones de los jugadores donde aparezca dicho número; y, finalmente, determinar los premios de “línea” y “bingo” correspondientes a una partida. Adicionalmente, se deberá repetir el juego para el número de partidas o rondas prefijado, establecer un **número de puntos** por cantar “línea” y “bingo” en cada partida (por ejemplo, 10 puntos por línea y 30 puntos por partida), y determinar el jugador ganador absoluto, junto con su puntuación total, en una tanda de partidas de bingo (por ejemplo, se podría jugar una tanda de 5 partidas).

En la solución de la práctica habrá que diseñar e implementar en **Java**, al menos, los siguientes *Tipos Abstractos de Datos* (TADs) que se describen a continuación:

- **TAD TandaPartidas**: Representa una secuencia con un número de partidas prefijado (al comienzo del juego).
- **TAD Partida**: Representa una partida del juego del bingo.
- **TAD Bombo**: Describe el conjunto de bolas disponibles para una ronda del juego (inicialmente, son 90 bolas diferentes, cada una con un número entre el 1 y el 90). El bombo contiene las bolas que los jugadores podrían “tachar” (si fuese necesario) durante cada jugada.
- **TAD BolasAparecidas**: Representa la secuencia de bolas que han sido sacadas hasta el momento en una partida.
- **TAD Jugador**: Representa la información de cada jugador humano participante en una tanda de partidas de bingo. Entre esta información debería estar, al menos, el nombre del

jugador, los cartones que juega en la tanda de partidas, y los puntos acumulados durante (y al final) de la tanda.

- TAD *Cartón*: Describe los números contenidos en el cartón de un jugador. Dichos números se irán tachando del mismo conforme vayan apareciendo sus números en el avance de una partida. Es necesario que cada cartón tenga la capacidad de “pintarse” cuando sea necesario.
- TAD *Línea*: Representa la colección de números contenidos en una línea de un cartón de bingo. Es interesante, también, que cada línea de un cartón tenga la capacidad de “pintarse” cuando sea necesario.

Las reglas del juego “simplificado” del dominó serán las siguientes:

- 1) Jugarán una partida un número comprendido entre 2 y 4 jugadores “humanos”.
- 2) El juego consistirá en una secuencia (o tanda) de partidas de bingo cuyo número se prefijará al comienzo del juego. También se decidirá al comienzo la puntuación asignada a “cantar línea” y “cantar bingo” en la tanda completa de partidas. Por lo tanto, ganará el juego quien consiga una mayor puntuación en la tanda de partidas.
- 3) Inicialmente, se establecerán los números contenidos en el cartón asignado a cada jugador. Los cartones de los jugadores para cada partida se visualizarán al comienzo de la misma.
- 4) La puntuación asignada, al comienzo del juego, a cada jugador participante será de 0 puntos.
- 5) También, al comienzo del juego, la “secuencia de bolas aparecidas” (y en orden de aparición) estará vacía. Dicha secuencia (y la correspondiente estructura de datos) se irá actualizando en cada jugada conforme se vayan sacando bolas del bombo.
- 6) Una “jugada del bingo” consiste en “sacar una bola del bombo” por parte del ordenador. Ello conlleva los siguientes pasos:
 - Presentar en pantalla la bola sacada.
 - Quitar la bola del bombo (para no poder sacarla nuevamente) e incluirla en la estructura de “bolas aparecidas”.
 - “Tachar” dicha bola en los cartones de jugadores donde aparezca.
 - Presentar por pantalla la secuencia de “bolas aparecidas”.
 - Actualizar adecuadamente las estructuras de datos involucradas en dicha jugada.
- 7) Cuando, por primera vez durante una partida, se hayan tachado los 5 números de una línea contenida en un cartón de un jugador, esté cantará: “¡Línea!” y conseguirá el premio correspondiente. Esto conllevará los siguientes pasos (adicionales a los correspondientes de “sacar bola”):
 - Visualizar el nombre del jugador (o jugadores) que ha(n) “cantado línea”, una vez verificada la línea cantada en el cartón del jugador.
 - Presentar por pantalla, los números de la línea cantada (en orden de aparición).
 - Sumar a la puntuación del jugador(es) premiado(s) los puntos asignados a “cantar línea”.
- 8) Cuando en una partida se hallan tachado todos los 15 números contenidos en el cartón de un jugador, esté cantará: “¡Bingo!” y se acabará dicha partida. Esto conllevará los siguientes pasos (adicionales a los correspondientes de “sacar bola” y “cantar línea”):

- Visualizar el nombre del jugador que ha “cantado bingo”, una vez verificado el cartón de bingo del jugador ganador.
 - Presentar por pantalla, los números del premio del bingo (en orden de aparición).
 - Sumar la puntuación asignada al cartón de bingo al jugador correspondiente.
- 9) Una vez acabada una tanda de partidas de bingo prefijadas, se determinará el “ganador absoluto” de la tanda. Esto conllevará los siguientes pasos (adicionales a los correspondientes para “sacar bola”, “cantar línea” y “cantar bingo”):
- Presentar por pantalla el número de puntos totales del jugador ganador (y felicitarlo).
 - *Opcionalmente*, se podría visualizar por pantalla una lista ordenada en orden de decreciente con las puntuaciones de los jugadores en la tanda de partidas (escribiendo en cada línea de pantalla los datos de cada jugador).

Un ejemplo concreto.

A continuación, se muestra la posible salida parcial por pantalla (que contiene solo algunas de las bolas sacadas del bombo) para una partida ejemplo. En ella, todo es automático excepto la entrada manual de los datos iniciales correspondientes a: 1) el número de partidas de la tanda, 2) el número de jugadores, 3) los nombres de los jugadores, y 4) los puntos asignados a “cantar línea” y “cantar bingo” en la tanda de partidas del juego (dichos datos aparecen en rojo y en cursiva en el ejemplo).

```
JUEGO DEL BINGO
=====

Introducir el número de partidas a jugar:
> 1
Introducir el número de jugadores:
> 2
Nombre Jugador 1:
> Juan
Nombre Jugador 1:
> Marta
Puntos por cantar línea:
> 10
Puntos por cantar bingo:
> 30

¡Comienza la ronda de 1 partidas!

PARTIDA 1:

Cartón "Juan":
Línea 1: * 15 * 32 * 52 61 * 82
Línea 2: * * 20 * 40 54 * 72 90
Línea 3: 7 * * 38 44 * 68 75 *

Cartón "Marta":
Línea 1: 1 14 * * * 54 64 * 81
Línea 2: * * 20 * 46 58 66 79 *
Línea 3: 3 16 21 32 * * * * 87

Jugada 1:
Se saca del bombo la bola: 37
Bolas sacadas: 37
No la tiene ningún jugador
```

Jugada 2:
Se saca del bombo la bola: 64
Bolas sacadas: 37 64
La tiene: Marta

Jugada 3:
Se saca del bombo la bola: 20
Bolas sacadas: 37 64 20
La tiene: Juan Marta

.....

Jugada 9:
Se saca del bombo la bola: 72
Bolas sacadas: 37 64 20 90 40 54 61 52 72
La tiene: Juan
¡Línea de Juan!
Números de la línea (en orden de aparición): 20 90 40 54 72

.....

Jugada 26:
Se saca del bombo la bola: 21
Bolas sacadas: 37 64 20 90 40 54 61 52 72 58 14 21 66 87 46 14 32 54 79
81 1 16 88 3 11 21
La tiene: Marta
¡Bingo de Marta!
Números del bingo (en orden de aparición): 64 20 58 14 21 66 87 46 32
54 79 81 1 16 3 21

Ganador ronda de 1 partidas de bingo: Marta, ¡enhorabuena!
Puntos finales de Marta : 30
Puntos finales de Juan : 10

Observaciones sobre la solución implementada:

- 1) El ejemplo anterior está incompleto. Las líneas escritas con puntos suspensivos indican que algunas jugadas intermedias no se han escrito (pero aparecerían en pantalla como salida de las jugadas de la partida en curso). En dicho ejemplo, se han pintado con caracteres '*' las posiciones de los mismos donde no hay contenidos números (huecos).
- 2) En la solución planteada se valorará la elección adecuada de las estructuras de datos para representar cada uno de los TADs del juego.
- 3) Es posible (y deseable) utilizar la API de la librería *Java Collections* en la solución de esta práctica, cuando resulte posible.
- 4) Se valorará también la claridad de la solución planteada, junto con la complejidad algorítmica de la misma.
- 5) *Se valorará muy positivamente la generación automática de "cartones válidos" para los jugadores, siempre respetando la estructura establecida para los mismos (véase el ejemplo de cartón válido de la Fig. 1).*
- 6) Por último, también se valorará positivamente cualquier otra mejora incluida en la solución implementada (no descrita en el enunciado), y que respete los requisitos establecidos para la práctica. Todo lo que no esté concretado en este enunciado puede ser decidido por el alumno libremente, indicándolo claramente en la memoria entregada.