



**Examen #1 de Cátedra**  
II-2019  
Miércoles 9 de octubre del 2019

**Profesores:**

- Francisco Arroyo
- Edgar Casasola
- Ricardo Gang
- Maureen Murillo
- Javier Vásquez

**Observaciones generales:**

1. Cuenta con 3 horas para entregar su solución del examen.
2. El examen es individual. Es prohibido utilizar herramientas digitales para intercambio de documentos.
3. Guarde su celular. Si lo utiliza para lo que sea, se considerará fraude.
4. Firme la hoja de asistencia al finalizar el examen.

**Observaciones específicas para examen realizado en papel:**

1. Puede utilizar el material escrito previamente autorizado por su profesor.

**Observaciones específicas para examen realizado en computadora:**

1. Puede utilizar cualquier material y código propio escrito por usted (prácticas, tareas, libros, apuntes) y material de Internet debidamente referenciado. Si no cumple con estas características se considerará fraude.
2. Como nombre de su proyecto utilice "ex1" junto con su carnet y nombre. Por ejemplo: ex1B82345JoseAraya.
3. En el comentario principal de cada clase, como autor indique su número de carné y su nombre.
4. Cada 20 minutos suba su solución en un archivo comprimido que contenga el código fuente al sitio de entrega del examen de su grupo.
5. En caso de fallo de energía eléctrica debe continuar su examen en papel. Se calificará la versión subida hasta ese momento a la página del grupo, así como la documentación que aporte en su cuaderno de examen.
6. La solución del examen debe subirse dentro del tiempo asignado a la página usada en su grupo. Al finalizar su examen y por motivos de seguridad guarde en su cuenta de correo, archivo drive o memoria USB una copia de la solución entregada.

## Enunciado BUSCA 13

Programe una aplicación que permita jugar a dos personas (identificados por su nombre). Se tiene un tablero lineal con  $n$  casillas ( $n$  es un valor entre 12 y 16, que debe ser definido por el usuario al inicio del juego). Cada casilla del tablero posee un valor aleatorio (enteros entre 0 y 3) y un color aleatorio (verde, amarillo o rojo).

Cada jugador posee 3 fichas colocadas inicialmente en los extremos del tablero.

El juego lo inicia el jugador definido en primer lugar y alternadamente cada jugador escoge entre:

1. mover una ficha indicando cuál ficha y hacia donde (casilla libre) moverla
2. finalizar la partida

Para obtener el puntaje se suma el valor de las celdas del jugador y posteriormente se aplican, de izquierda a derecha, las modificaciones asociadas a los colores. El color verde duplica el valor del puntaje que se está calculando, el amarillo no lo altera y el rojo le resta una unidad. El objetivo del juego es lograr que el puntaje sea igual a 13.

En todo momento debe mostrársele al usuario el estado del tablero y el puntaje de cada jugador en una sola ventana. El programa debe informar al usuario el ganador tan pronto alguno de los dos sume 13.

Ejemplo de contenido. No asuma que se debe mostrar así.

0	1	2	3	4	5	6	7	8	9	10	11
a	v	a	r	r	v	r	v	a	v	v	a
0	1	3	3	1	2	1	0	2	3	0	2
Paco	Paco		Lola	Paco	Lola	Lola					

Puntaje de Paco = 3

//cálculo :  $(0+1+1)*1*2-1$

Puntaje de Lola = 9

//cálculo :  $(3+2+1)-1*2-1$

Para la solución de este problema **NO** puede utilizar ninguna clase del API para manejo de colecciones, como por ejemplo *ArrayList*.

### Lineamientos generales de evaluación

- a) Cumplimiento de buenas prácticas de programación.
- b) Separación de responsabilidades de los objetos.
- c) Correcta especificación de clases.
- d) Uso correcto de constructores.
- e) Uso correcto de la programación orientada a objetos.
- f) Uso de arreglos de objetos.
- g) Uso correcto de estructuras de control.

h) Entrada/salida.

**Evaluación específica**

- 1) 45% Definición correcta del modelo de clases y manejo de datos (setters, getters, constructor, toString)
- 2) 45% Correcta funcionalidad del juego
- 3) 10% Convenciones: correcta nominación, indentación