

Materia:	<b>Programación I</b>		
Nivel:	<b>1º Cuatrimestre</b>		
Tipo de Examen:	<b>Segundo Parcial</b>		
Apellido <sup>(1)</sup> :		Fecha:	<b>27 nov 2024</b>
Nombre/s <sup>(1)</sup> :		Docente a cargo <sup>(2)</sup> :	<b>Fernández / Pavlov</b>
División <sup>(1)</sup> :	<b>212</b>	Nota <sup>(2)</sup> :	
DNI <sup>(1)</sup> :		Firma <sup>(2)</sup> :	

(1) Campos a completar solo por el alumno.

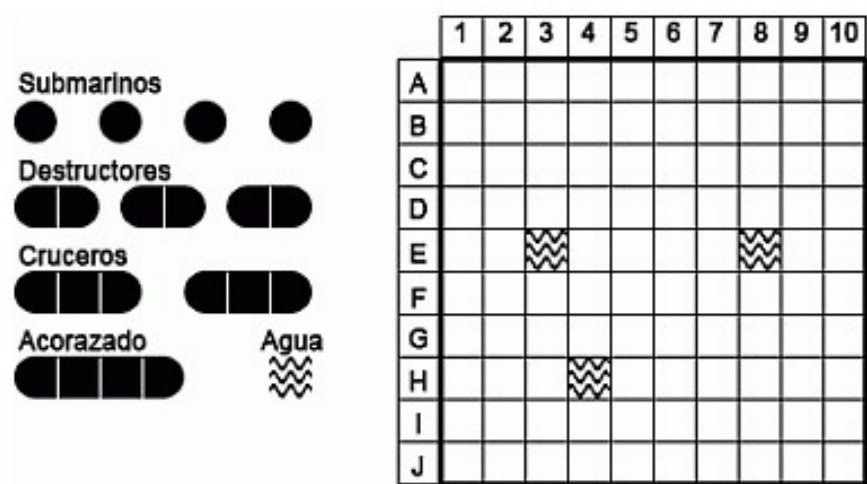
(2) Campos a completar solo por el docente.

(3) Las instancias válidas son: 1º Parcial (**P1**), Recuperatorio de 1º Parcial (**RP1**), 2º Parcial (**P2**), Recuperatorio de 2º Parcial (**RP2**), Recuperatorio Integrador (**RIN**), Final (**F**), Recuperatorio de Final (**RF** - Solo válido para seminario de nivelación). Marcar lo que corresponda con una cruz.

### Desarrollar en Python:

Videojuego **Batalla Naval**, que tendrá los siguientes datos:

- Matriz de 10 filas por 10 columnas.
- Datos para llenar la matriz de 10 filas por 10 columnas.
- Datos para colocar en la matriz de 10 filas por 10 columnas cada una de las naves.
- Dificultad:
  - o Nivel Fácil: matriz de 10 filas por 10 columnas.
  - o Nivel Medio: matriz de 20 filas por 20 columnas y duplicar la cantidad de naves.
  - o Nivel Difícil: matriz de 40 filas por 40 columnas y triplicar la cantidad de naves.



## Requerimientos:

A. Para el estado inicial: desarrollar una función que realice la creación dinámica de una matriz de 10 filas por 10 columnas. En la misma se deberá incluir ceros para el agua y unos consecutivos para las naves. Las naves pueden ser horizontales y/o verticales. Las naves deben ser generadas de la siguiente manera:

- Cuatro (4) **submarinos** de un (1) casillero
- Tres (3) **destruidores** de dos (2) casilleros
- Dos (2) **cruceros** de tres (3) casilleros
- Un (1) **acorazado** de cuatro (4) casilleros

B. Crear una pantalla de inicio, con cuatro (4) botones:

- **Nivel**
- **Jugar**
- **Ver Puntajes**
- **Salir**

La pantalla deberá tener:

- Una imagen cubriendo completamente el fondo.
- Sonido de fondo.

Al hacer clic en el botón **Jugar** se iniciará el juego.

C. En la pantalla del juego:

- Habrá un tablero donde a cada casillero del mismo le corresponderá un elemento de la matriz.
- Crear un (1) botón con la etiqueta **Reiniciar**.

D. Al comenzar el juego se deberá imprimir el **Puntaje** en **0000** donde se irá decrementando en un (1) punto por cada disparo errado (pudiendo tener puntaje negativo). Cada disparo acertado sumará cinco (5) puntos, al hundir la nave se obtendrán diez (10) puntos extras por cada elemento que compone la misma.

Por ejemplo: Si la nave es de tres (3) elementos consecutivos de la matriz, sumará 30 puntos, además de los cinco (5) puntos sumados anteriormente por haber averiado cada uno de los elementos que la componen.

E. Al hacer clic en uno de los casilleros del tablero se efectuará el disparo. No se podrá efectuar el disparo sobre el mismo casillero más de una vez.

F. Al disparar:

- Si el disparo es acertado (da en el blanco):
  - Si la nave es averiada, se debe sumar cinco (5) puntos.
  - Si la nave es hundida, se debe sumar diez (10) puntos por cada elemento que contenga la misma.
- Si el disparo hace agua (no da en el blanco), se debe restar un (1) punto.

G. Al hacer clic en el botón **Reiniciar** se reiniciará el juego, y el puntaje volverá a estar en **0000**.

- H. Antes de comenzar o una vez terminado el juego se deberá pedir el nombre al usuario (Nick), guardar ese nombre con su puntaje en un archivo, y volver a la pantalla de inicio.
- I. Al ingresar a la pantalla inicial y hacer clic en el botón **Ver Puntajes**, se deberá mostrar los 3 (tres) mejores puntajes ordenados de mayor a menor, junto con su nombre de usuario (Nick) correspondiente. Debe haber un (1) botón para volver al menú principal.

#### OPCIONALES:

- Nivel de dificultad.
- Pantalla de inicio: agregar un (1) botón para **activar** / **desactivar** el sonido de fondo.
- Al disparar:
  - Reproducir un sonido de disparo acertado.
  - Reproducir un sonido de disparo errado.
- Agregar imágenes, sonidos, y animaciones donde corresponda.

---

#### Condiciones de aprobación:

- **Aprobación NO Directa:**
  - Deberá estar realizado en su totalidad los ítems A, B, C, D, E, F, G.
- **Aprobación Directa:**
  - Deberá estar realizado en su totalidad los ítems A, B, C, D, E, F, G, H e I. (**OPCIONALES** solamente sumará puntos si están en condiciones de Aprobación Directa).

#### NOTAS:

**Nota 0:** Se deberá desarrollar biblioteca y funciones propias, las mismas deberán estar correctamente documentadas.

**Nota 1:** El set de datos debe generarse dinámica y aleatoriamente (la distribución de las naves deberá ser diferente en cada partida).

**Nota 2:** El formato del archivo que se debe crear para guardar puntaje y nick tiene que ser TXT, CSV o JSON.