

**Universidad Tecnológica Nacional
Facultad Regional Avellaneda**



Técnico Superior en Programación - Técnico Superior en Sistemas Informáticos

Materia: Laboratorio de Programación II

Apellido:		Fecha:	06/08/2020
Nombre:		Docente ⁽²⁾ :	
División:	2º	Nota ⁽²⁾ :	
Legajo:		Firma ⁽²⁾ :	
Instancia ⁽¹⁾ :	<input type="checkbox"/> PP <input type="checkbox"/> RPP <input type="checkbox"/> SP <input type="checkbox"/> RSP <input type="checkbox"/> X <input type="checkbox"/> FIN <input type="checkbox"/> X		

(1) Las instancias validas son: 1º Parcial (PP), Recuperatorio 1º Parcial (RPP), 2º Parcial (SP), Recuperatorio 2º Parcial (RSP), Final (FIN). Marque con una cruz.

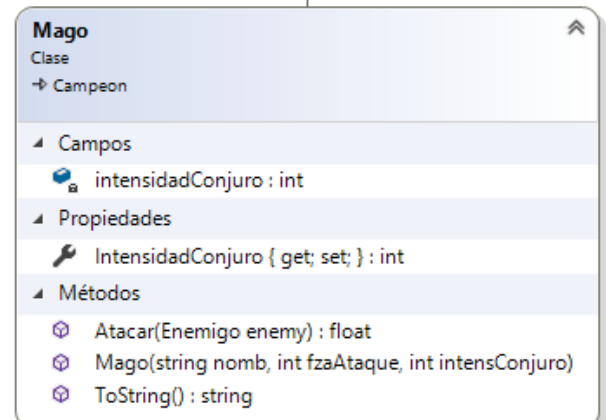
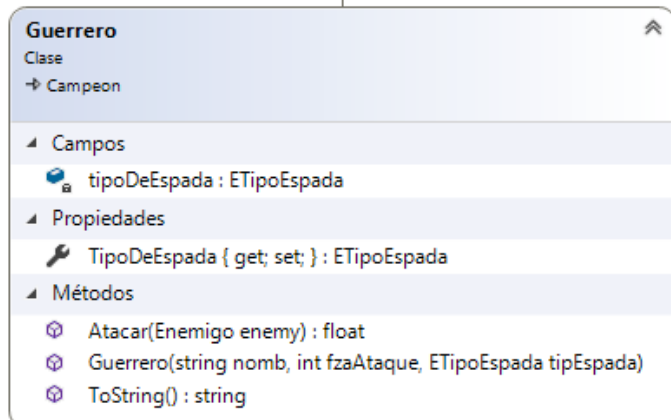
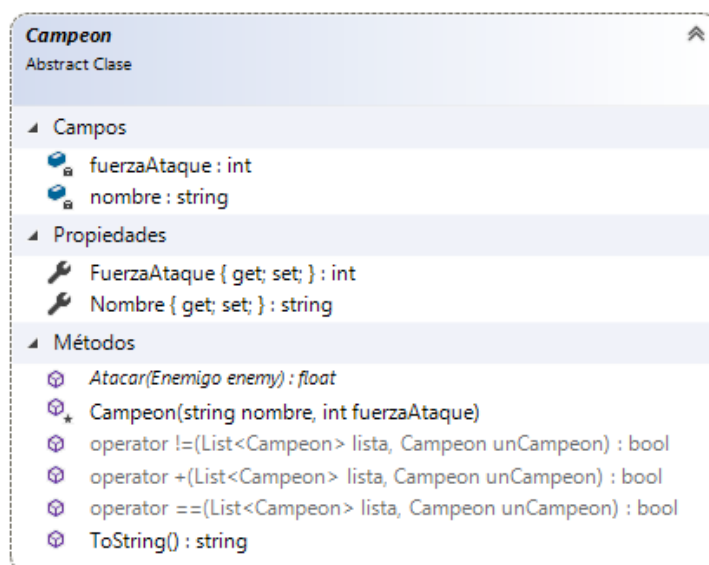
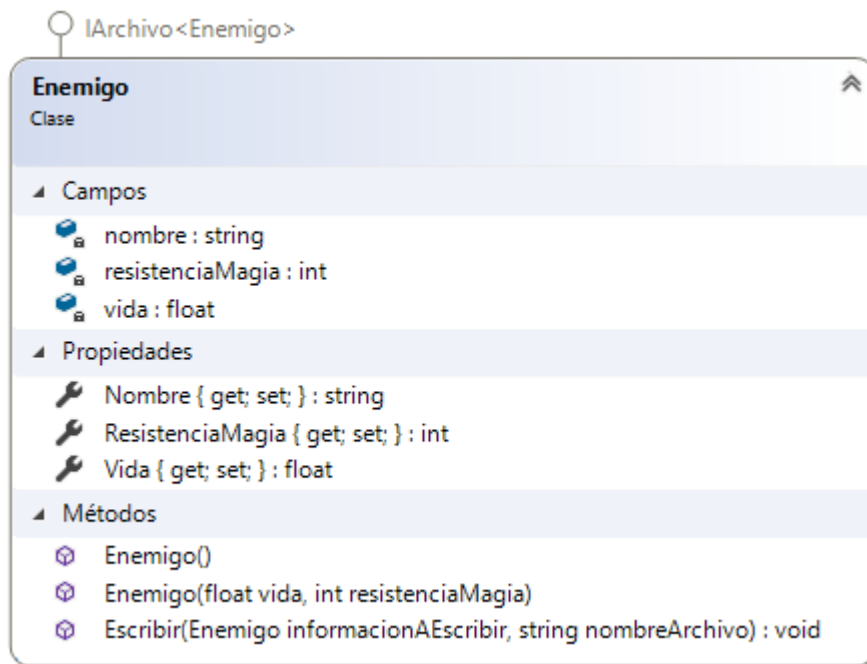
(2) Campos a ser completados por el docente.

IMPORTANTE:

- **2 (dos) errores en el mismo tema anulan su puntaje.**
- **La correcta documentación y reglas de estilo de la cátedra serán evaluadas.**
- Modificar el nombre de la solución y colocar sus datos personales de la siguiente forma: Apellido.Nombre. Ej: Pérez.Juan.2D. No se corregirán proyectos que no sea identificable su autor.
- **TODAS** las clases deberán ir en una Biblioteca de Clases llamada Entidades.
- No se corregirán exámenes que no compilen.
- **Reutilizar** tanto código como crean necesario.
- Colocar nombre de la clase (en estáticos), **this** o **base** en todos los casos que corresponda.
- Colocar el nombre del docente con el cual cursó.

TIEMPO MÁXIMO PARA RESOLVER EL EXAMEN 90 MINUTOS.

- Se pide desarrollar una primera beta de un videojuego al estilo RPG. Esta primera fase consistirá en desarrollar la lógica que permita que dos personajes ataquen a un enemigo hasta eliminarlo.
 - Aclaración: Tanto la serialización xml como la escritura de archivo de texto, deberán guardarse en el path relativo MIS DOCUMENTOS. El nombre del archivo queda a criterio del alumno.
 - Programar sobre el proyecto adjunto. No crear otra solución nueva.
- Generar un proyecto con el nombre del alumno del tipo Windows Form y una biblioteca de clases llamada Entidades.
- La biblioteca de clases tendrá el siguiente formato:



Extension
 Static Class

Métodos

CalcularBonificacion(this Campeon champion, Enemy enemy) : float

IArchivo<string>

Batalla
 Clase

Campos

enemy : Enemy
 hiloBatalla : Thread
 listaHeroes : List<Campeon>

Métodos

Batalla()
 ComenzarBatalla() : void
 CrearEnemy() : void
 CrearHeroesIniciales() : void
 Escribir(string informacionAEscribir, string nombreArchivo) : void
 IniciarHilo() : void

Eventos

eventoStatus : DelegadoMensaje

Tipos anidados

DelegadoMensaje
 Delegado

mensaje : string

ETipoEspada
 Enum

Katana
 Cimitarra

CampeonYaExistenteException
 Clase
 Exception

Métodos

CampeonYaExistenteException(string message)

IArchivo<T>
 Genérico Interfaz

Métodos

Escribir(T informacionAEscribir, string nombreArchivo) : void

Enemigo

- En el constructor por defecto asignará “El Malote” como nombre del enemigo.
- Implementará el método Escribir de la interfaz IArchivo. Serializará un enemigo en el test unitario.

Campeón

- Clase abstracta.
- Sobrecarga Operador ==
 - retornará true si el Campeón ya existe en la lista.
 - Un campeón será igual a otro cuando su nombre y su tipo sean iguales.
 - (Tip: pueden utilizar la función GetType para saber el tipo del objeto).
- Sobrecarga Operador +
 - Permitirá agregar un nuevo Campeón a la lista de campeones de la clase Batalla, siempre y cuando el mismo NO exista en la lista (Reutilizar código).

Mago

- Atacar: retornará el resultado la siguiente operación matemática:
 - `this.FuerzaAtaque * this.intensidadConjuro * this.CalcularBonificacion(enemy);`
- ToString: Retornará toda la información del objeto.

Guerrero

- Atacar: retornará el resultado la siguiente operación matemática:
 - `this.FuerzaAtaque * this.CalcularBonificacion(enemy);`
- ToString: Retornará toda la información del objeto.

Metodo de Extensión CalcularBonificacion

- Extenderá la clase Campeón
- Retornará un flotante
- Recibirá por parámetro un enemigo.
- Si el campeón recibido es un mago, se deberá preguntar si la resistencia a magia del enemigo es menor a la intensidad del conjuro del mago. Si lo es, la función devolverá 1.20f.
- Si el campeón recibido es un guerrero y su tipo de espada es una katana, devolverá como bonificación 1.25f.
- Si el campeón recibido es un guerrero y su tipo de espada es una cimitarra, devolverá como bonificación 1.15f.
- Si no es ninguno de los casos anteriores, devolverá 1.

Batalla

- En el constructor por defecto se instanciará la lista de héroes y el thread hiloBatalla. A este último se le asociará el método ComenzarBatalla.
- CrearHeroesIniciales(): Copiar y pegar el siguiente código:

```
Mago unMago = new Mago("Mago Pepe", 21, 76);

if (listaHeroes + unMago)
{
    eventoStatus.Invoke("Cargado el mago " + unMago.ToString());
}

Guerrero unGuerrero = new Guerrero("Romeo Gatuso ", 50, ETipoEspada.Katana);

if (listaHeroes + unGuerrero)
{
    eventoStatus.Invoke("Cargado el guerrero " + unGuerrero.ToString() );
}
```

- CrearEnemigo(): instanciar el enemigo ENEMY con 5000 de vida y 70 de resistencia a magia.
- ComenzarBatalla:
 - Mientras que la vida del enemigo sea mayor a 0, recorrerá la lista de héroes atacando al enemigo. Para ello, se deberá restarle al enemigo el resultado de ataque del héroe llamando al método Atacar.
 - Utilizar el método ESCRIBIR de esta clase para poder registrar quien y cuánto daño le ha hecho al enemigo en un archivo de texto.
 - Una vez eliminado el enemigo, abortar el hilo e invocar el eventoStatus enviando como parámetro el siguiente mensaje "Simulación Finalizada, enemigo derrotado".

Windows Forms

- Cambiar el título del formulario: Debe quedar de la siguiente forma "Simulador batallas - dev : Apellido.Nombre".
- Reemplazar Apellido y nombre por los datos del alumno.

Unit Test:

- Desarrollar un proyecto de unit testing que contenga las siguientes funcionalidades:
 - Serialice un Enemigo a Xml.
 - Capturar la excepción CampeonYaExistenteException al agregar un nuevo Guerrero que ya existe a una lista de guerreros (utilizar sobrecarga ==).