

# Algoritmos y Programación II

1C 2020

## Trabajo Práctico 2

El trabajo práctico 2 consiste en diseñar un TDA Ahorcado, luego implementarlo en una clase y generar un *main* que lo utilice.

La clase Ahorcado debe construirse con un string a adivinar (de tres letras o más) y un entero mayor o igual que 3 que es la cantidad de fallos que el jugador puede tener sin perder el juego.

El TDA debe diseñarse en un documento según el ejemplo de la sección 3.3 del documento teórico (apunte unificado). Además, hacer el diagrama UML de clase.

La palabra a adivinar deberá almacenarse en un vector dinámico de caracteres. El resto de la implementación queda a criterio de cada uno.

El juego comienza mostrando guiones en el lugar de cada letra que forma la palabra. El jugador puede:

- ir arriesgando de a una letra. La aplicación colocará las letras en el lugar correspondiente si hubiera alguna coincidencia, de lo contrario incrementará en uno la cantidad de fallos cometidos indicando cuántos intentos fallidos le quedan.
- arriesgar un string completo, por lo que el método arriesgar deberá estar sobrecargado. En caso de no acertar restará dos intentos.

El juego finaliza cuando el jugador adivina la palabra ya sea completando las letras o arriesgando la palabra entera, o cuando agota los intentos permitidos.

Además, en el constructor deberá imprimir la dirección inicial del vector de caracteres haciendo lo mismo en el destructor.

### A tener en cuenta

- Separación de archivos (h y cpp)
- Funcionalidad
- Interfaz con el usuario
- Nombres de métodos y variables
- Pre y post condiciones
- Modularización
- Utilización de memoria dinámica
- Uso correcto del sangrado
- Diseño

### Entrega

La fecha de entrega límite es el lunes 18 de mayo a las 23.55hs por el campus (se entregan los archivos en un solo archivo comprimido).