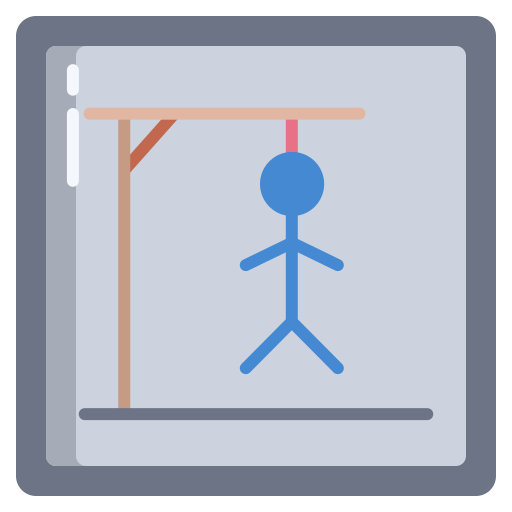
# **Juego del Ahorcado**



### **Idea general**

Desarrollar el juego de “**El ahorcado**” en consola. El objetivo del juego es que el jugador adivine una palabra secreta relacionada con una temática, letra por letra. Tendrán una cantidad limitada de intentos y podrán obtener puntos en función de aciertos y errores.

### **Funcionamiento del Juego**

El juego se desarrollará de la siguiente forma:

**Selección de palabra**:

Al inicio de cada partida, el programa elegirá una palabra al azar relacionada con una categoría específica (**programación, historia, deportes, videojuegos**)

Ejemplo:

* Temática: Deportes
* Palabra a descubrir: **fútbol**
* Palabra oculta mostrada al jugador: **\_ \_ \_ \_ \_ \_**

**Adivinanza de letras**:

* El jugador ingresa una letra.
* Si la letra es parte de la palabra, se muestra por consola en la posición correcta de la palabra oculta.
* Si la letra no está en la palabra, se le resta un intento y se informa por consola los intentos restantes. Son 7 intentos en total (cabeza, torso, cada pierna, cada brazo y la horca)

**Puntuación**:

* El jugador **ganará 3 puntos** por cada **letra adivinada** correctamente.
* Por cada **intento fallido**, **perderá 1 punto.**
* Si adivina la palabra completa, acumulará el puntaje de la partida en su puntuación total.

### **Requerimientos**

El programa debe estar debidamente estructurado según ciertos criterios:

1. **Funciones (mínimas)**:
   * **obtener\_lista\_palabras()**:obtendrá una lista de palabras desde un archivo csv.
   * **jugar\_ahorcado()**: permitirá que el jugador juegue la cantidad de veces que desee
   * **seleccionar\_categoria()**: elegirá de manera aleatoria una de las posibles categorías.
   * **seleccionar\_palabra()**: elegirá una palabra aleatoria de la lista de palabras.
   * **mostrar\_palabra\_oculta()**: mostrará la palabra oculta con guiones bajos para las letras no descubiertas.
   * **actualizar\_palabra\_oculta()**: tomará una letra ingresada y actualizará la palabra oculta y los intentos restantes.
   * **calcular\_puntuacion\_parcial()**: calculará y actualizará el puntaje basado en aciertos y errores en una partida.
   * **verificar\_estado\_juego()**: determinará si el jugador ganó o no la partida.
   * **calcular\_puntuacion\_final()**: calculará y actualizará el puntaje basado en los puntajes parciales de cada partida.
   * **guardar\_puntuacion()**: guardará el nombre del jugador, junto con la puntuación total y la fecha, en un archivo csv.

A

1. **Paquetes y módulos:** el programa debe estar adecuadamente dividido en paquetes y módulos, en virtud de los contenidos vistos en clase.