Algoritmos y estructuras de datos

**Trabajo Práctico Nº 3**

**El ahorcado**

Desarrolle un programa para jugar al popular juego del ahorcado, dicho juego consiste en un personaje ficticio al cual hay que salvar de ser ahorcado.

**El primer jugador** cargará una palabra, la misma no podrá tener espacios, puede utilizar las funciones para sacar espacios de izquierda y derecha. Una vez cargada la palabra se debe borrar la pantalla y empieza el segundo jugador.

**El segundo jugador** para salvarse debe adivinar la palabra, de la cual sólo se conoce su longitud. El jugador debe ir eligiendo letra por letra, de modo de ir completando la palabra o arriesgar con toda la palabra.

Si el jugador se equivoca en una letra, es decir, la letra seleccionada no pertenece a la palabra a adivinar, se va formando un muñeco con partes del cuerpo (un brazo, una pierna, el tronco, etc).

Se debe mostrar en la pantalla al segundo jugador: el muñeco con sus partes, la palabra con \_ y las letras que ya adivinó, cantidad de intentos, cantidad de errores y el listado de las letras que ya ingreso.

Se puede jugar hasta los 7 intentos erróneos o hasta descubrir la palabra.  
Si hizo el intento con toda la palabra, es este caso pierde o gana sin importar en qué número de intento lo hizo.  
Si ingresa una letra que ya se cargó anteriormente debe avisar y permitir que vuelva a cargarla sin tener en cuenta este ingreso como erróneo.

**Observaciones:**

* El programa de hace en grupo de 2 personas
* La idea de este trabajo práctico es que dividan tareas entre los participantes del grupo. Deberían dividirse las funciones, desarrollarlas por separado y documentarlas para que el otro integrante sepa cómo se deben usar.  
  Terminadas las funciones deben hacer en común el cuerpo principal del algoritmo utilizando las funciones realizadas por cada uno de los integrantes.
* Debe utilizar una matriz de 2 filas para el manejo de las palabras, una fila para la palabra original y la otra fila para la palabra que se muestra con las letras que fue adivinando.
* Utilice un vector de control de letras para almacenar cada letra ingresada por el segundo jugador.
* Cuando muestre la palabra al segundo jugador se debe asignar el carácter “\_” a cada letra no adivinada de la palabra y cada vez que se adivine una letra, substituya el carácter “\_” por la letra correspondiente en la posición que va.
* **El programa se debe resolver utilizando como mínimo estas 4 funciones.** 
  1. Que verifique que el ingreso de un dato de tipo caracter sea una letra. De A a Z debe incluir la Ñ como válida.
  2. Que verifique que la palabra ingresada por el primer jugador no tengan espacios intermedios y que sean solo letras.

Esta función también se debe aplicar si el segundo jugador arriesga la palabra completa.

* 1. Que verifique la longitud del ingreso. Que lo ingresado por el segundo Jugador tenga un solo carácter o la cantidad total de caracteres de la palabra.
  2. Que verifique la letra ingresada no exista en el vector de letras ingresadas.
* **El programa se debe resolver utilizando como mínimo estos 2 procedimientos**.
  1. Que busque la letra ingresada en la matriz:

(Acá tener en cuenta parámetros por Referencia)

-Si la encuentra debe reemplazar en la fila de la palabra que muestra el “\_” por dicha letra. Tener en cuenta que puede estar más de una vez,

- Si no la encuentra debe actualizar el contador de cantidad de errores sumándole uno.

* 1. Que realice el dibujo del ahorcado.