

# Práctica I

Se desea realizar un juego del ahorcado. Puede haber cualquier tipo de palabra. La palabra se escogerá aleatoriamente de un archivo de texto (que el programa debe leer) formado por una lista de 10 palabras.

## Paso 1 (2 puntos)

Crear un programa que pida al usuario una letra. El programa de indicar si esa letra forma parte de la palabra elefante o no. Si la letra aparece varias veces indicará cuántas veces aparece y sus posiciones.

## Paso 2 (3 puntos)

El jugador va a tener 7 vidas.

Si cuando el usuario introduce la letra no forma parte de la palabra, debe decrementar una variable, llamada "vidas", cuyo valor inicial es 6. Cada vez que el usuario introduzca una letra el programa debe decir:

la letra yyy forma parte que la palabra secreta, te quedan xxx vidas

o bien

la letra yyy no forma parte que la palabra secreta, te quedan xxx vidas

## Paso 3 (3 puntos)

Para mostrar lo que el usuario ha adivinado trabajaremos con una variable tipo *string* que llamaremos adivinada: `std::string adivinada{"*****"};` que inicialmente serán todo asteriscos (y tiene el mismo número de asteriscos que letras tiene la palabra).

Cada vez que el usuario introduzca una letra:

1. Comprobar si esa letra pertenece a la palabra.
2. En caso de que pertenezca ver su posición o posiciones.
3. Comprobar el valor que hay en esa posición en la variable *adivinada*.
4. Si hay un \* sustituir el \* por la letra adivinada.

## Paso 4 (2 puntos)

Dibujar por pantalla el monigote ahorcandose cada vez que va perdiendo vidas.



```
  |
  0
 /|\
 / \
```

## NOTA

La estructura del programa es la siguiente:

```
#include <string>
#include <iostream>

int main(){
    std::string palabra{"avion"};
    std::string adivinada{"*****"};
    int vidas{7};
    while(vidas > 0){

        // EL PROGRAMA VA AQUI

        if(adivinada == palabra){
            std::cout << "Has ganado\n";
            return 0;
        }
    }

    std::cout << "Oh, oh\n\n";
    std::cout <<
    " |\n" <<
    " 0\n" <<
    "/|\\n" <<
    "/ \\n";

    return 0;
}
```