

# JUEGO SERPIENTES ( MONSTRUOS) Y MAZMORRAS EN PYTHON

Queremos diseñar en Python, un juego RPG-Rol con estas normas:

Al comenzar elegimos nombre de personaje y nuestra habilidad ( una de las 3 posibles: lucha,magia,astucia ), automáticamente y de forma aleatoria se nos asignara una puntuación ( vida ) desde el comienzo, con un valor entre 100 y 200 puntos.

Nuestro objetivo en el juego es llegar a la sala de FIN con al menos 1 punto de vida, en ese caso y tras derrotar al monstruo final, habremos ganado la partida.

Se perderá la partida en el momento de que perdamos todos nuestros puntos de vida.

Cada sala puede tener un máximo de 4 puertas ( se nombran según los puntos cardinales Norte,Sur,Oeste,Este ), Cada vez que se entra en una sala , se nos indica cuales son sus posibles salidas, un mensaje de ambiente, los posibles objetos que existen y el posible monstruo que en sala habita.

Solo se puede avanzar por las salas, nunca puedes volver hacia atrás , hacia una sala en la que ya has estado.

En un sala solo hay 4 acciones posibles: ir a otra sala, recoger objeto,luchar si hay monstruo,huir si hay monstruo.

Los objetos en el caso de que lo queramos recoger de la sala, pueden subir puntos,bajar puntos o no hacer nada en nuestra habilidad para el combate con los monstruos.No todas la salas tienen objetos y como máximo puede existir 2 objetos no iguales por sala, la probabilidad de que haya al menos un objeto es de un 75% y de 2 objetos al menos de un 30%.

Cuando se recoge un objeto no se sabe que habilidad nos refuerza ni el numero de puntos que tiene, solo se sabe en el momento que finalice la lucha cuando se usen para la lucha.

Los monstruos no tienen por que estar en todas las salas ( en la sala FIN, si existe de forma obligatoria ) , existe un 75% de posibilidades que exista un monstruo en un sala, si en la anterior sala que hemos estado ya existía un monstruo, en la actual solo existe un 66% de que exista otro.

Cuando en una sala existe un monstruo, podemos elegir luchar o salir huyendo a otra sala contigua, en el caso de que queramos salir huyendo , se nos restara 50 puntos por ser un cobarde.

Para luchar, una vez elegido el comando de lucha, entramos en lucha con el monstruo que este en la sala, para luchar disponemos de 3 dados con puntuación del 1 al 20 y el monstruo ( el ordenador lucha como el monstruo ) 2 dados con puntuación de 1 a 25.

Se lanzan de forma aleatoria nuestros 3 dados y los 2 dados del monstruo.

Se elige automáticamente los 2 dados de mayor puntuación nuestros y se comparan con la puntuación de los 2 dados del monstruo, si nuestra puntuación es igual o mayor en ambos dados, hemos ganado la lucha, si

solo es mayor o igual en 1 de los dados, es empate y se repite la tirada, si los 2 del monstruo son mayores a los 2 nuestros, hemos perdido la lucha.

En el caso de que ganemos la lucha se nos sumara los puntos de victoria del monstruo y en caso de que perdamos se nos restara esos puntos.

Uso de los objetos en la lucha: ( se elige el que queremos usar, solo uno cada vez en cada lucha, no es obligatorio elegir objeto para la lucha, cuando se usa un objeto en la lucha, se pierde y no se puede volver a utilizar en la partida ).

En el caso de perder la lucha, se nos restara de vida, los puntos acorde a la puntuación del monstruo

Cada objeto que podemos recoger de una sala puede sumar o restar puntos en el caso de victoria en una lucha.

Si el objeto que hemos elegido es de la misma habilidad ( de las 3 posibles ) que la habilidad del monstruo, se nos sumara los puntos extra sobre el valor de los puntos obtenidos en la victoria.

Si el objeto que hemos elegido no es de la misma habilidad ( de las 3 posibles ) que la habilidad del monstruo, se nos restara los puntos extra sobre el valor de los puntos obtenidos en la victoria.

Si el objeto que hemos elegido es de la misma habilidad ( de las 3 posibles ) que la habilidad del monstruo y también igual que nuestra habilidad como personaje se nos sumara un 20% extra sobre el valor de los puntos obtenidos en la victoria.

Cada vez que se efectúa una lucha, el sistema nos indica si ganamos o perdemos y el numero de puntos implicados.

El mensaje al entrar en una sala con monstruos y objetos puede ser:

***"Estas en una mazmorra helada y fría, tan solo se escucha el sonido del hielo, tienes 2 objetos en ella una espada y unas canicas, existe un monstruo, que se llama víbora, tienes salidas al norte y al oeste" "#"***

Mensaje de lucha:

**"has ganado la lucha contra víbora usando una espada, has ganado 50 puntos"**

A través de un prompt (#) indicamos al juego que queremos hacer

***Los comandos del juego son:***

**N,S,E,O para ir a una sala contigua.**

**I para mostrar los objetos que tenemos.**

**P para mostrar los puntos de vida que tenemos.**

**L para luchar.**

**J para luchar usando un objeto.**

**H para huir sin luchar.**

**R para recoger un objeto.**

**A para ayuda y mostrar los comandos disponibles.**

El Juego Se basa en 4 ficheros que son totalmente personalizables, para que tengamos la posibilidad de jugar siempre a partidas distintas.

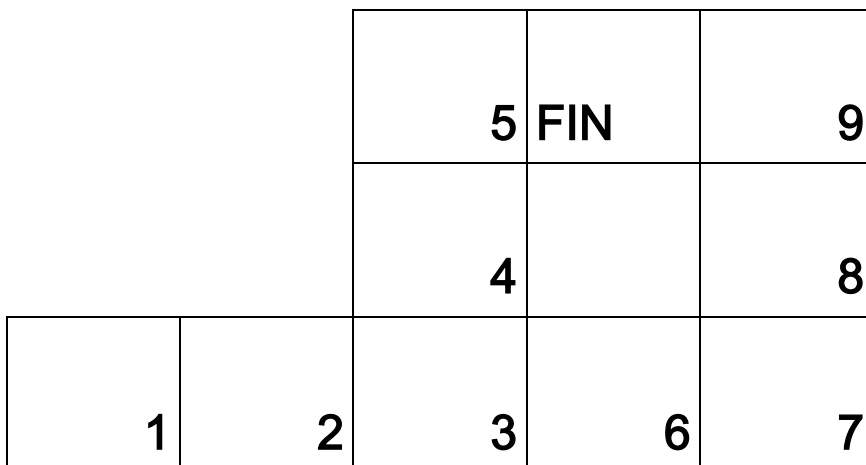
Los 4 ficheros son:

**mapa.txt**

**monstruos.txt**

**objetos.txt**

**ambiente.txt**



CAMINO1 para llegar a la ultima sala: 1,2,3,4,5,FIN

CAMINO2 para llegar a la ultima sala: 1,2,3,6,7,8,9,FIN

Ejemplos del contenido de los ficheros de gestión del juego;

Fichero **mapa.txt** ,N°SALA COORDENADAS SALA VECINA N S O E

1;0;0;0;2  
2;0;0;1;3  
3;4;0;2;6  
4;5;3;0;0  
5;0;4;0;FIN  
FIN;0,0;5;9

Fichero **monstruos.txt** ,N° MONSTRUO,NOMBRE, PUNTUACION INICIAL,HABILIDAD

1;PITON;100;MAGIA  
2;VIBORA;50;LUCHA  
3;COBRA;75;INVISIBILIDAD

Fichero **objetos.txt** ,NOMBRE,HABILIDAD,PUNTUACION

1;UNA ESPADA;LUCHA;10  
2;UNA VARITA;MAGIA;20  
3;UNA CAPA DE INVISIBILIDAD;MAGIA;30  
4;UN MARTILLO;LUCHA;20  
5;UN LIBRO DE POCIMAS MALDITO;MAGIA;15  
6;UNAS CANICAS;ASTUCIA;12

Fichero **ambiente.txt**

1;Estas en una mazmorra helada y fría, tan solo se escucha el sonido del hielo  
2;La sala es oscura, tenebrosa y sin luz  
3;En la cueva, solo suena el leve susurro del fuego y del terror

# EVALUACIÓN

El programa completo se debe de entregar como muy tarde el 17 de diciembre, teniendo que haber una primera entrega parcial como muy tarde el 26 de noviembre, esta entrega parcial no es necesario incluir la gestión de luchas , ni las opciones extras si las tuviese, dejando estas 2 tareas para la entrega final.

Entregar en formato comprimido todos los ficheros necesarios para su ejecución, si se usa librerías no estándares, usar el fichero requirements.txt para incluirlas.

## **Versión texto: máximo 6 puntos**

### **Sobre 6 puntos:**

3 puntos:

Coherencia del código, estructura, uso de funciones y módulos, uso de comentarios, documentación de funciones y parámetros.

2 puntos:

Ejecución acorde a las especificaciones del documento, validaciones sin errores en tiempo de ejecución,

1 punto:

Vistosidad gráfica, originalidad en la inclusión de novedades tecnológicas en el código.

## **Versión texto + Version TKinter: máximo 8 puntos**

### **2 puntos:**

Uso apropiado de los elementos gráficos de TKinter, validaciones, documentación del código, vistosidad gráfica.

## **Versión texto + Version TKinter+Extras(solo texto): máximo 10 puntos**

### **2 puntos:**

1 punto:

Añadir la posibilidad de ver la sala , realizada con caracteres ASCII, al estilo del ejemplo que se muestra.

Para ello codificar un nuevo modulo: gestiongraficos.

Posibilidad de crear los 4 ficheros mediante una opción dentro del programa, Para ello codificar un nuevo modulo: gestionficheros.

1 punto:

Posibilidad de guardar y cargar el estado de una partida, con nuevos comandos en el Prompt: G: grabar C: cargar Para ello codificar un nuevo módulo : gestionpartidas.

Posibilidad de al empezar la partida, elegir el grado de dificultad: bajo, medio o alto, influirá en las posibles puntuaciones de partida tanto del jugador como del monstruo, así como el sistema de puntuación de los dados y los objetos, Para ello codificar un nuevo módulo : gestiondificultad.

=====

Ejemplo de Dibujo de una SALA

```
=====N=====
[   X               ]
[                   ]
[           Z       ]
[                   E]
[       *   0       ]
[       * - | -     ]
[       /   \       ]
=====S=====
```

N,E,S Son las posibles Entradas/Salidas de la Sala

X y Z Son Objetos

```
*   0
*-|-   Es un Monstruo
/   \
```

Ver mas ejemplos en:

<https://blog.desdelinux.net/en/I%27m-a-terminal-geek-and-my-favorite-games-are-roguelike/>

## INSTRUCCIONES PARA CODIFICAR

-Lo ideal es trabajar con múltiples módulos que se puedan importar ( import ), de forma que exista:

- gestionficheros : para gestionar todo lo relacionado con los ficheros.
- gestionsalas: para gestionar todo lo que se realizar en una sala.
- gestionlucha: para gestionar las luchas con los monstruos.
- gestionobjetos: para gestionar todo lo relacionado con los monstruos
- main : modulo principal del juego.

Para el uso de los ficheros, a todos los efectos se pueden tratar como ficheros CSV, ver ejemplo en la Moodle de como trabajar con ellos.

Si se guarda una partida, lo ideal es también guardar los 4 archivos de uso del programa, junto con los datos de la partida, es posible que cuando queramos recuperar una partida grabada, ya se hayan modificado los datos de algunos de los 4 archivos y haya inscosistencia de los datos.