

Seminario de Lenguajes - Python

Trabajo final

Enunciados

Opción 1: Programando sopas de letras

Objetivo

Desarrollar dos aplicaciones:

- **SopaProgramada**: un juego de programación para niño/as pequeño/as, y
- **SopaProgramadaConf**: una aplicación para el/la docente, mediante la cual pueda configurar algunos aspectos del juego.

Objetivo general del juego:

Encontrar una palabra en una sopa de letras “programando” al personaje para que la resalte en la grilla.

Objetivos específicos: introducir los conceptos básicos de instrucción, programa y secuencia.

Destinatarios

Los destinatarios de **SopaProgramada** serán niño/as de nivel primario que comienzan a trabajar aspectos básicos de programación.

Los destinatarios de **SopaProgramadaConf** serán lo/as docentes a cargo de lo/as niño/as.

Descripción

SopaProgramada será un juego de programación presentado en varios niveles de complejidad donde los jugadores deberán ubicar y resaltar una palabra de forma similar al juego [Concurso de Ortografía](#) propuesto por la iniciativa code.org. La Fig. 1 muestra una captura del nivel inicial del mismo.



Fig. 1: Nivel inicial de code.org

Al igual que en este juego, un personaje recorrerá la sopa de letras y resaltaré las letras correspondientes a la palabra encontrada. La palabra a encontrar puede o no estar visible en alguna zona de la interfaz, dependiendo de cómo se configuró el juego. Para lograr el objetivo de cada nivel, el jugador deberá indicar los movimientos a través de órdenes o comandos que formarán un programa sencillo.

Las instrucciones iniciales serán: Avanzar - Retroceder -- Girar a Derecha - Girar a Izquierda. Al desplazar (avanzar y retroceder) el personaje, irá marcando los casilleros por donde pasa, resaltando la letra correspondiente. Esto significa que en el primer nivel la palabra a resaltar comienza en un celda adyacente a la posición inicial del personaje. Las instrucciones de girar a derecha y a izquierda no desplazan al personaje, sólo lo “apuntan” hacia otro lado.

El juego permitirá que el jugador pueda eliminar la última instrucción agregada.

Cada nivel introducirá una complejidad adicional:

1. Los dos niveles iniciales mostrarán una única palabra que comienza en un casillero adyacente a la posición inicial del personaje.
2. Los dos niveles siguientes incorporarán una nueva instrucción: `salto_casillero`, que permitirá que el personaje avance un casillero sin resaltar la letra del mismo. Esto permite que la palabra se encuentre en cualquier lugar de la grilla.
3. Los dos niveles siguientes permitirán incorporar parámetros a las instrucciones: por ejemplo Avanzar 3 (acá se podría mostrar un menú desplegable con opciones fijas, por ejemplo, 1, 2 o 3) o Salto_casillero 3.

El juego deberá presentar al menos todos los niveles propuestos e incluir opciones de configuración básica tales como habilitar/deshabilitar efectos de sonidos y música, atajos de teclas, mostrar o no la palabra a encontrar, etc.

Las palabras se obtendrán de un archivo previamente generado por la/os docentes con la aplicación **SopaProgramadaConf**.

SopaProgramadaConf será una aplicación destinada a lo/as docentes de manera tal que puedan generar las posibles palabras a esconder en la sopa de letras. Las palabras se agruparán en categorías propuestas por lo/as docentes. Por ejemplo, podrán agrupar palabras en: sustantivos y verbos o por temas: astronomía, matemáticas, personajes de cuentos, etc. En todos los casos, SopaProgramada podrá mostrar no sólo la palabra en su versión original indicada por el/la docente, sino derivaciones de la misma, Por ejemplo: Si la palabra es SOL, podrá aparecer SOL o SOLES, y si la palabra es SALIR, podrá aparecer SALIR, SALIENDO o SALIMOS.

La idea es que el/la docente genere los archivos correspondientes y luego los copie en las computadoras donde lo/as alumno/as jugarán.

Opción 2: Elige tu propia historia...

Objetivo

Realizar una aplicación denominada **CuentoPersonalizado** que visualice cuentos interactivos, permitiendo al lector decidir el rumbo que la historia toma en distintos momentos.

Destinatarios

Los destinatarios serán niño y niñas entre 6 y 99 años.

Descripción

CuentoPersonalizado se basa en los libros de texto “Elige tu propia aventura”, donde el lector, en algunos puntos específicos de la historia, toma decisiones que lo conducen a distintos desenlaces.

CuentoPersonalizado presentará los cuentos interactivos disponibles para leer. Luego de elegir uno de ellos, debe procesar y representar los guiones del mismo. Éstos se encuentran en archivos de texto que respetan un formato específico¹ en un directorio denominado **cuentos**.

En la historia habrá diferentes escenas, actores y diálogos. La descripción de los mismos se encuentran en los archivos que contienen los guiones para cada historia posible.

Escenas

Una escena es un fragmento de la historia en donde suceden eventos.

¹ En la parte final de este documento se presenta un ejemplo del mismo.

Los **guiones**, estarán divididos por escenas. La forma en que aparecerá la información de la escena en el archivo del guión, es la siguiente:

```
...
[playa, playa.png]
```

Ocupará una línea dentro del archivo y aparecerá entre corchetes. El primer elemento es el nombre de la escena y, el segundo, el nombre del archivo que contiene el fondo que se debe mostrar.

Lo que corresponda a una escena abarca desde la aparición de la línea [nombreEscena, archivoFondoEscena], hasta la definición de otra escena o hasta el final del archivo.

El orden de lo que suceda en la escena estará dado por el orden de aparición en el guión de los elementos correspondientes de la misma..

La primer escena será siempre la inicial.

Actores y diálogos

Un actor es un personaje con un nombre y sus diálogos en distintos momentos de la historia. Los actores deberán decir su diálogo según el orden de aparición en el guión. Los diálogos se definirán como una secuencia líneas, donde cada línea respetará el siguiente formato: nombre del actor, **dos puntos** (separador) y diálogo (sin saltos de línea).

Ejemplo:

```
...
Ackbar: Es una trampa!
```

El actor de nombre Ackbar dirá “Es una trampa!” en el momento dado por el orden de aparición en la escena.

Las imágenes de los actores se encuentran en un directorio con el nombre del cuento y el archivo correspondiente será el nombre del actor.

La posición inicial del actor será una al azar y será otorgada la primera vez que aparezca el actor en la historia.

Acciones

Los actores podrán realizar acciones. La información de la acción del actor aparecerá en una línea del guión de la siguiente forma:(nombre del actor, nombre de la acción, parámetros necesarios para la acción). Por ejemplo, para que el actor Luis camine hasta la posición x=100, y=150:

```
...
(Luis, caminar, 100, 150)
...
```

Las acciones que podrán realizar los actores son:

- “**posicionar**”, parámetros: x, y. El actor aparece (se posiciona, sin animación) en la posición x, y.
- “**caminar**”, parámetros: x, y. El actor se mueve a paso lento hasta la posición x, y.
- “**correr**”, parámetros: x, y. El actor se mueve a paso rápido hasta la posición x, y.
- “**saltar**”, parámetros: x, y. El actor se mueve saltando hasta la posición x, y.
- “**volverse_loco**”, sin parámetros. El actor gira sobre su eje 3 veces.
- “**reir**”, sin parámetros. El actor emite el sonido de risa y sobre su imagen aparece un emoticón de risa.
- “**seguir_a**”, parámetros: actor2. El actor se mueve hasta la posición del actor2.
- “**llorar**”, sin parámetros. El actor emite el sonido de un llanto y sobre su imagen aparece un emoticón de tristeza/llanto.
- “**hablar_infinitivo**”, sin parámetros. Todos los siguientes diálogos del actor serán reemplazados por sólo los verbos en infinitivo.
- “**hablar_normal**”, sin parámetros. Si el actor hablaba en infinitivo, sus siguientes diálogos volverán a la normalidad.
- “**sonar**”, parámetro: nombre de archivo de sonido. Deberá sonar el sonido o música indicado por el nombre de archivo.

Se deberán proponer 3 nuevas acciones para los actores documentando su propósito y forma de uso.

Decisiones

Son preguntas que se le harán al lector al finalizar una escena y que, en base a su respuesta se determinará el curso de la historia. Cada respuesta tendrá una escena asociada a la cual la historia saltará si elige esa opción. Las decisiones se definirán **entre llaves**: primero estará la pregunta, luego *dos puntos* (separador), y las posibles opciones separadas por *punto y coma*. Las opciones se escribirán como: texto de la opción, igual (signo separador) y nombre de la escena².

Por ejemplo, “Carlos se va de viaje. A dónde debería ir?” París o Roma:

...

```
{Carlos se va de viaje. A dónde debería ir?:París=EscenaInicioRoma;Roma:EscenaInicioParís}
```

...

Si no existe una decisión al finalizar la escena, la historia finaliza.

Calificación de la historia

Al finalizar la historia, se le pedirá al jugador que califique la historia con 0, 1, 2, 3, 4 o 5 estrellas, guardando el puntaje acumulado de la historia.

Al iniciar **CuentoPersonalizado** se mostrarán las historias disponibles con las estrellas otorgadas por quienes leyeron ya la historia.

Configuración

Se deberán poder configurar los siguientes valores:

- Sonido: Activar o desactivar
- Tiempo de duración de los diálogos.

² El nombre de la escena es el que se indica en el ítem *Escenas*

Esta información deberá guardarse en un archivo y cargarse al iniciar la aplicación.

Aclaraciones

- No se limiten sólo a lo que está definido, si agregan mejoras, el juego será mejor (y la nota también!)
- Pueden usar el “posicionar” para ubicar a los actores inicialmente en la escena, al igual que el “sonar” para reproducir música y sonidos.
- Si quieren poner un “salto de línea” en el diálogo de un actor, dividan el diálogo en dos. Por ejemplo, Carlos dice:

*“Hola,
Cómo andan”*

Escríbanlo como:

Carlos: Hola

Carlos: Cómo andan

- Los actores de cada escena deberán obtenerse de acuerdo a los definidos tanto en los diálogos como en las acciones.

Ejemplo de un guión simple en formato texto

[Escena1,fondo1.png]

actor1: Hola

(actor1,caminar,100,150)

actor2: Como andas?

auto: Broom! Broom!

{Que debe hacer el actor1?:responder=Escena2;no responder:Escena3}

[Escena2,fondo1.png]

actor1: Bien

(auto,correr,100,150)

actor2: Cuidado!

actor1: Casi me pisa ese auto.

actor2: Zafaste

[Escena3,fondo1.png]

(actor1,correr,200,150)

(auto,correr,100,150)

actor2: Que rapido que pasan los autos

actor1: Esta complicado

Consideraciones generales

Para ambos trabajos:

- Usar PIlasEngine para las interfaces gráficas.
- Pueden definir y utilizar cualquiera de los formatos de archivos vistos en la cursada para los archivos que requieran generar.
- Usar el módulo pattern.es en el caso que consideren necesario.

- Documentar el código usando docstrings en las funciones.
- Incluir en cada archivo fuente los nombres de los autores y la licencia.
- Armar una estructura de directorios para organizar los códigos en carpetas y subcarpetas de manera tal que sea claro su corrección.
- Incluir en el directorio raíz entregado un archivo denominado **leer.txt** que incluya el nombre de los integrantes, la forma de ejecutar cada aplicación y cualquier consideración especial para su ejecución (por ejemplo, funciona en sistemas Windows o Linux).
- Usar recursos libres o propios que puedan ser publicados con licencias libres.
- Entregar en un único archivo con formato **zip**.

Fechas de entrega

Primera Entrega: Semana del

Segunda Entrega: Semana del

Criterios de evaluación

- Funcionalidad implementada de acuerdo a enunciados.
- Defensa oral individual.
- Respetar las consideraciones planteadas.

Recursos libres

Algunos enlaces que les pueden ser útiles..

Tipo de recurso	Sitio
Imágenes	<ul style="list-style-type: none"> • https://openclipart.org/ • https://www.flickr.com/
Música y Sonidos	<ul style="list-style-type: none"> • https://www.jamendo.com/ • http://opsound.org
Videos	<ul style="list-style-type: none"> • https://vimeo.com/ • https://www.youtube.com • https://commons.wikimedia.org/wiki/Category:Videos
Tipografías	<ul style="list-style-type: none"> • http://ospublish.constantvzw.org/foundry/ • https://fonts.google.com/ • https://www.theleagueofmoveabletype.com/
Varios	<ul style="list-style-type: none"> • https://www.nasa.gov • https://www.openstreetmap.org/ • Internet Archive: https://archive.org/ • http://publicdomainreview.org. • https://commons.wikimedia.org • https://creativecommons.org/about/platform