

Introducción a la programación

TutiFrutiUNGS...

May 7, 2020



Introducción

El trabajo consiste en implementar un juego para entrenarse en el TutiFruti. La idea es que el jugador o jugadora mejore sus tiempos y aprenda nuevas palabras para que cuando enfrente a sus amigos y amigas logre sorprenderles. En pantalla aparece una letra y un item, por ejemplo "V" y "Colores" entonces el jugador debe escribir un color que empiece con dicha letra, en este caso podría ser "violeta". Gran parte del juego ya está resuelto, solamente faltan implementar las funcionalidades más importantes.

1 El Juego

Reglas del Juego

Aparecerán en pantalla una letra y un item, el jugador va eligiendo los nombres, países, animales, etc y al finalizar el juego se muestran sus elecciones y una elección correcta que hizo la computadora.

Lo que ya está implementado

El juego actualmente consta de un archivo con el programa principal. Este se encarga de capturar la entrada del teclado, llevar la cuenta de los puntos y del tiempo, así como también de dibujar en la pantalla.

El programa principal cuenta también con: Tres listas, una con colores, otra con países y otra con animales. Otra lista con los nombres de estos items y una lista de listas llamada `listaDeTodo` que contiene como primer elemento la lista de colores, luego la de países y por último la lista de animales. Podría ser de esta manera por ejemplo: `[["azul", "celeste", "marron"], ["argentina", "uruguay", "venezuela", "brasil"], ["mono", "jirafa", "marmota", "mula"]]`

Para la mayoría de estas tareas, el programa hace uso de una biblioteca de código llamada *PyGame*. Una biblioteca de código es un conjunto de sub-programas utilizados para desarrollar software. En particular PyGame es una biblioteca especialmente diseñada para el desarrollo de juegos interactivos en Python.

La posición $(0, 0)$ de la pantalla es el vértice superior izquierdo, las x crecen hacia la derecha y las y crecen hacia abajo.

Lo que falta implementar

Aun faltan implementar las funciones del archivo `funciones.py`. Estas funciones son utilizadas desde el programa principal.

La función `unaAlAzar(lista)`: recibe una lista, el abecedario por ejemplo, y devuelve una letra al azar.

La función `esCorrecta(palabraUsuario, letra, item, items, listaDeTodo)` recibe una palabra, una letra, un item, la lista de items y la listaDeTodo, y devuelve un puntaje positivo si la palabra es correcta y uno negativo en caso contrario. Se considera correcta solo si la palabra aparece en el item correcto y empieza con la letra correcta.

La función `juegaCompu(letraAzar, listaDeTodo)` dada una letra y la lista de listas devuelve una lista con una palabra al azar de cada item que empiece con esa letra, en caso de no tener palabra para algún item devolverá una cadena vacía en esa posición de la lista. por ejemplo: si la letra es "m" la lista de `items=[["azul", "celeste", "marron"], ["argentina", "uruguay", "venezuela", "brasil"], ["mono", "jirafa", "marmota", "mula"]]` entonces podría devolver `["marron", "", "mono"]`

2 Cómo empiezo

2.1 Instalar PyGame

La versión más reciente de PyGame para Windows y Python 3.1 se descarga directamente desde <http://pygame.org/ftp/pygame-1.9.1.win32-py3.1.msi>. para diferentes versiones, chequear <http://www.pygame.org/download.shtml>. La versión de python instalada recomendamos que sea de 32 bits.

si tienen problemas pueden:

1. Descargar pygame y luego este archivo desde <https://www.lfd.uci.edu/~gohlke/pythonlibs/#pygame>

Pygame : una biblioteca para escribir juegos basada en la biblioteca **SDL** .

[pygame - 1.9.6 - cp38 - cp38 - win_amd64.whl](#)
[pygame - 1.9.6 - cp38 - cp38 - win32.whl](#)
[pygame - 1.9.6 - cp37 - cp37m - win_amd64.whl](#)
[pygame - 1.9.6 - cp37 - cp37m - win32.whl](#)
[pygame - 1.9.6 - cp36 - cp36m - win_amd64.whl](#)
[pygame - 1.9.6 - cp36 - cp36m - win32.whl](#)
[pygame - 1.9.6 - cp35 - cp35m - win_amd64.whl](#)
[pygame - 1.9.6 - cp35 - cp35m - win32.whl](#)
[pygame - 1.9.6 - cp27 - cp27m - win_amd64.whl](#)
[pygame - 1.9.6 - cp27 - cp27m - win32.whl](#)
[pygame - 1.9.4 - cp34 - cp34m - win_amd64.whl](#)
[pygame - 1.9.4 - cp34 - cp34m - win32.whl](#)

2. Ejecutar cmd como administrador e ir a la carpeta donde hayan guardado el archivo.
3. Escribir `py -3.7 -m pip install pygame-1.9.6-cp37-cp37m-win32.whl`

2.2 Descargar archivos

Descargar del moodle de la materia el archivo comprimido con todos los archivos necesarios para el TP. Descomprimir todo el contenido del archivo en una carpeta y abrir los archivos `.py` con el PyScripter. El único archivo que se espera que modifiquen es `funciones.py` pero a la hora de ejecutar el proyecto, hay que ejecutar el archivo `principal.py`.

3 Consigna

Implementar las funciones requeridas para el correcto funcionamiento del juego.

Pensar e implementar funciones auxiliares que resuelvan tareas intermedias, de forma tal que el código sea más claro, sencillo, ordenado, legible y fácil de corregir.

Las funciones que reciben listas como parámetros deberán también chequear que dichas listas permanezcan en el estado correcto luego de utilizada la función.

Sugerimos **fuertemente** probar y corregir las funciones más sencillas antes de encarar las funciones más complicadas.

4 Requisitos de aprobación

El presente trabajo debe realizarse en grupos de 2 o 3 integrantes. Para aprobar el trabajo, se deberá cumplir los siguientes items:

- El juego debe funcionar correctamente.
- El código debe ser claro. Es decir, las variables y funciones deben tener nombres que hagan fácil de entender el código a quien lo lea, y deben haber comentarios que ayuden al fácil entendimiento de cada porción de código. Además debe hacer funciones siempre que se considere necesario, y se evaluará el buen uso de las mismas.
- El código debe ser coherente. Es decir, no deben haber variables que no se usan, funciones que tomen parámetros que no necesitan, ciclos innecesarios, etc.
- Los casos de prueba deben ser completos y debe ser posible correrlos de nuevo por cualquiera.

Nota: El correcto funcionamiento del juego no es suficiente para la aprobación del trabajo, son necesarios todos los items mencionados arriba.

5 Opcionales

Las siguientes funcionalidades del juego no son necesarias para la aprobación (con nota mínima), pero sirven para mejorar la nota del trabajo. De optar por hacerlas, se aplican las mismas reglas y criterios de corrección que para las funcionalidades básicas. Cualquier otra funcionalidad extra que se desee implementar debe ser antes consultada con los docentes.

Lectura

Tienen que hacer la lectura de archivos y cargarlos en las listas. El archivo puede ser uno solo con marcas que indiquen donde termina un item y comienza otro. O pueden

Más items

Extender la cantidad de items, por ejemplo, capitales, frutas y verduras, nombres, etc.

Más puntaje

Puntaje especial si la computadora elige la misma palabra para un ítem.

Efectos de sonido

Hacer que el juego reproduzca efectos de sonido cuando sucedan los eventos más importantes o errores.

6 Fecha de entrega

El trabajo debe ser entregado en la fecha estipulada en el cronograma, recordar que es requisito hacer pre-entregas.

7 Forma de entrega

El TP se considera correctamente entregado y apto para ser corregido si se cumple con una entrega del informe y del tp funcionando correctamente.

- La entrega consiste de un **informe** donde haya una introducción que explique de qué se trata el trabajo (explicado para alguien que no leyó el enunciado), que incluya el código de las funciones implementadas y una breve explicación de cada una de ellas junto con las dificultades de implementación con las que se encontraron. El informe también debe incluir las decisiones que hayan tomado ante diferentes alternativas posibles, cuáles fueron sus alternativas consideradas y por qué tomaron esas decisiones. El informe **no** debe incluir párrafos para ocupar lugar, explicaciones de funcionalidades básicas de Python, PyGame o de programación en general.
- La entrega digital consiste en enviar un e-mail de la siguiente manera:
 1. A la cuenta de correo electrónico de los docentes de la comisión.
 2. **Con asunto:** entrega-tp 2020 *nombre – estudiante*.
 3. Adjuntando por separado un archivo comprimido con los archivos que conforman el código fuente de **todo** el programa y un archivo con el informe (preferentemente en formato pdf). El código debe cargar sin errores o no será corregido.

Nota: Se acepta una sólo entrega por grupo, de modo que si un grupo entrega su trabajo más de una vez, los docentes elegirán cuál de las versiones entregadas corregir.