Informe de Introducción  a la Programación

Alumnos: Alcaraz Sofia, Matiazzo Francisco.

Profesores: Velcic Fernando, Nieto Agustin

Comisión: 03           banda horaria: 18:00-22:00

**Introducción:**

El siguiente trabajo consiste en hacer un videojuego en el cual se debe adivinar el orden de aparición de un subtítulo tomados de  la película de los simpsons.

Se juega de a un jugador, éste solo cuenta con 60 segundos para sumar la mayor cantidad de puntos. El jugador deberá escribir una palabra que se encuentre dentro de la línea del subtítulo que se ubica luego de que aparece primero en pantalla.

La mayoría del juego ya estaba hecho, nosotros nos encargamos de realizar las funcionalidades más importantes que se encuentran en un archivo del programa principal, el cual capta la entrada del teclado, lleva la cuenta de los puntos y del tiempo, también se encarga de dibujar en la pantalla.

A demás se nos brinda un archivo llamado “TheSimpson” con los subtítulos de la película de los Simpson, el cual debimos respetar y limpiar.

Para realizar estas tareas el programa  utiliza una biblioteca llamada pygame (diseñada para el desarrollo de videojuegos interactivos de python).

Para que el juego funcione uno de los archivos importantes que debimos completar fue el llamado funcionesVACIAS.py, que será explicado más adelante.

Por último también se nos brindó otros archivos importantes, los cuales son: principal (), configuración() y extras().

**FUNCIONES VACIAS:**

En cuanto al archivo “funcionesVACIAS.py”, dentro de esta se encuentran cuatro funciones imprescindibles para el funcionamiento del juego, es decir, función lectura (), función selección (), función puntos () y función procesar ().

**Función lectura:**

La función lectura recibe un archivo y un subtítulo (tipo lista, vacio) y retorna una lista con los subtítulos de menor longitud de N, es decir, 3 (este y otros datos se encuentran en configuración) y filtra el archivo quitando los caracteres especiales y limpiándolo.

Para poder completarla utilizamos la función “filtrar” y “caracteres\_especiales”.

Tuvimos dificultades con el enunciado, donde habla de al menos longitud N, pero luego de entenderlo pudimos hacer andar el juego.

También luego de la prueba del juego nos aprecia un carácter al final, que nos dimos cuenta de que no lo sacamos es decir “\n”, pero lo solucionamos quitando el ultimo carácter de líneas.

def lectura(archivo, subtitulo,N):

lineas=archivo.readlines()

i=2

while i < len(lineas):

while lineas[i] != "\n"and len(lineas)>=N:

subtitulo.append(filtrar(lineas[i][:-1]))

i=i+1

i=i+3

print (subtitulo)

return subtitulo

“Filtrar” recibe un subtitulo y devuelve el subtitulo ya limpio llamando a "caracteres\_especiales”.

def filtrar(linea):

subfinal=""

for letra in linea:

subfinal=subfinal+caracteres\_especiales(letra)

return subfinal

“Caracteres\_especiales” recibe cada letra del subtitulo de filtrar y las devuelve sacando los caracteres especiales, si no tiene un carácter “especial”, la devuelve como está.

def caracteres\_especiales(letra):

if letra=="á":

return "a"

if letra=="é":

return "e"

if letra=="í":

return"i"

if letra=="ó":

return "o"

if letra=="ú":

return"u"

if letra=="ñ":

return "n"

return letra

**Función selección:**

La función selección recibe  la lista subtítulos (ya limpios) y retorna una lista con un subtítulo al azar, su siguiente y otro. Las dificultades que surgieron con esta función fueron que el random otro no sea igual a el primero y su siguiente, pero lo arreglamos con un while del que solo puede salir si es diferente a estos dos.

def seleccion(subtitulo):

lista=[ ]

subtitulo\_random=random.randrange(0,len(subtitulo),2)

lista.append(subtitulo[subtitulo\_random])

lista.append(subtitulo[subtitulo\_random+1])

otro=random.randrange(0,len(subtitulo),3)

i=0

while otro==subtitulo\_random or otro==(subtitulo\_random+1) and i<len(subtitulo):

otro=random.randrange(0,len(subtitulo),3)

i=i+1

lista.append(subtitulo[otro])

return lista

**Función puntos:**

La función puntos recibe el número de seguidillas y devuelve el puntaje que alcanzó el jugador, según el número de seguidillas. Esta función es llamada por procesar.

def puntos(n):

return n+2

**Función procesar**:

La función procesar recibe la palabra ingresada por el usuario, un string, (palabraUsuario), la palabra mostrada (), la siguiente (), la otra () y la seguidilla de correctas. Esta función verifica que la palabra ingresada por el jugador se la correcta, es decir que pertenece a  siguiente y devuelve el puntaje según el número de seguidillas.

Para esta utilizamos una función aparte llama “stringalista” que se encarga de cambiar el subtítulo siguiente (cadena) a una lista.

La función procesar recibe la palabra ingresada por el usuario, un string, (palabraUsuario), la palabra mostrada (),la siguiente (), la otra () y la seguidilla de correctas. Esta función verifica que la palabra ingresada por el jugador sea correcta, es decir que pertenece a  siguiente y devuelve el puntaje según el número de seguidillas. A demás quita las mayúsculas.

def procesar(palabraUsuario, mostrada,siguiente, otra, correctas):

listasig=string\_a\_lista(siguiente)

puntaje=0

for palabra in listasig:

if palabra.lower()==palabraUsuario.lower():

puntaje= puntos(correctas)

return puntaje

Para esto llamamos una función aparte llamada “stringalista” que se encarga de cambiar el subtitulo siguiente (cadena) a una lista. Esta recibe una cadena y devuelve una lista. A demás quita los caracteres especiales(incluye espacios) que no son contemplados en procesar.

def string\_a\_lista(string):

cadena=""

lista=[ ]

for letra in string:

if letra != " " and letra!="!" and letra!="¡" and letra!="," and letra!="¿" and letra!="?" and letra!="." and letra!="<i>" and letra!="</i>" and letra!="\"":

cadena=cadena+letra

else:

lista.append(cadena)

cadena=""

lista.append(cadena)

listanueva=[ ]

for palabra in lista:

if len(palabra)!=0:

listanueva.append(palabra)

return listanueva