

INFORME DE ENTREGA FINAL PROGRAMACIÓN DECLARATIVA

Docente:

Danielle Valeria Carina

Alumnos:

De Aragón Juan Manuel

Moyano Lucas

Oliveto Jesús

Ingeniería Informática 2022

Universidad Blas Pascal

DEFINICIÓN DEL PROYECTO:

Motor de videojuego estilo Idle Game, con las funciones básicas para el desarrollo de uno, que sea capaz de adaptar su funcionamiento para un videojuego nuevo.

FUNCIONALIDAD:

La moneda principal es "r", la cual se utiliza para comprar mejoras cuya función es aumentar la velocidad del incremento de esta.

El incremento se ve reflejado en el "deltaR", que simboliza cuanto incrementa r por segundo.

r se calcula en base al cálculo $r = w * A + k$, donde w es el nivel de prestigio, A es el resultado de $A = a * b * c * d$, y k es la variable independiente, utilizada para incrementar r rápidamente en el juego temprano, perdiendo importancia a medida que A incrementa.

El juego admite la posibilidad de prestigiar, lo cual resetea los valores de las mejoras a como se definen por defecto, pero incrementando el valor de w a razón de +1 por cada 100 mejoras de A compradas, por lo que cada vez que se prestigie, r crecerá más rápido.

PARTE TÉCNICA:

ESTRUCTURA:

clicker.py: posee el main loop, encargado de generar los distintos hilos para procesamiento de variables, generar el display y actualizarlo. Además es el encargado de guardar y cargar el progreso automáticamente.

data_.py: cuenta con las variables que se usarán en la ejecución del programa, con la finalidad de permitir la modularización.

funciones.py: conformado por los algoritmos que se ejecutan en el main loop, aquí yacen las funciones adaptables para cualquier IdleGame.

savegame.txt: archivo de guardado del juego. Se genera vacío en la primera ejecución, se escribe al cerrarse el programa y se lee al ejecutarlo.

var.txt: guarda el estado actual de las variables auxiliares de la misma forma que el savegame, con el objetivo de que el cargado del archivo se realice más rápido y se eviten errores en el display.

FUNCIONES:

funciones.py:

- .loadGame(): lee el archivo savegame.txt y var.txt para obtener los valores almacenados en el guardado. Se debe ejecutar al abrirse el juego.

- .DrawText(): escribe texto en la ventana. Recibe como parámetros el texto a escribir, el color del texto, el color del fondo, las coordenadas X e Y, y el tamaño de fuente.

- .rectangle(): dibuja un rectángulo en el que se define el display dibujado, el color de fondo, las coordenadas X e Y, el ancho y el alto.

- .rincrement(): incrementa el valor de r(moneda principal) y los valores de A y deltaR para ser mostrados en pantalla. Cicla a 30 ticks por segundo.

- .getAux*(): obtiene los valores auxiliares para la compra de mejoras en base al total de r que posee el usuario, teniendo en cuenta el incremento del precio por compra. El valor auxiliar es utilizado para calcular el incremento del valor de la mejora y para el display de las mejoras comprables.

- .getAuxw(): obtiene el auxiliar para la compra del prestigio, el cual se calcula en base al total de mejoras ya compradas.

- .saveGame(): escribe los archivos savegame.txt y var.txt para almacenar las variables del juego al momento del cierre del mismo. Es lo último que se ejecuta.

- .prestige(): lleva a cabo el prestigio, lo cual se basa en resetear todas las variables a sus valores originales, y a la vez incrementando el valor de w en base a los valores de estos previos al prestigio.