



**Tecnológico
de Monterrey**

Proyecto Final del Curso TC1028

Autor:

Edgar Roann Santillan Bernal

A00572737

Introducción:

Empecemos con la parte de librerías, donde usamos Fraction para las fracciones. Le da el formato de fracción. Tenemos random para generar datos random. El time sirve para hacer un delay. El pandas es para leer archivos de excel. Después, sigue la función la parte donde necesitas introducir tu nombre para empezar con el código. Después viene el bucle infinito para que se repita las veces que sean necesarias para el menú y te puedes salir cuando tu quieras.

Una vez guardado tu nombre, nos pasamos a la opción uno. Donde tenemos una función que nos imprime que va a ver dos rounds de 10 preguntas. Tenemos los acumuladores de correcto e incorrecto. Las primera diez preguntas se despliegan mientras que las correctas sean menor a 10. En ese bucle tenemos un dato random genera datos de intervalos de uno al doce. El resultado, es la operación de hacer el ordenador con los valores que uno le da. Después, preguntamos la pregunta del millón, donde se guarda el resultado del usuario. Si el resultado y la pregunta son lo mismo aumenta el acumulador de correcto más uno, pero si la tienes mal, el acumulador correcto se le resta menos uno y te imprime el resultado correcto. Si tienes diez correctas te da la opción de si quieres pasar al nivel dos donde es suma de potencias. Y si no quieres seguir jugando te regresa al menú. En el nivel dos las potencias se hacen primero y después se suman. Si tienes 20 correctas completas el nivel, pero sino te dice cuantas tuviste mal y cuántas bien.

En la suma de fracciones tenemos dos niveles donde se genera primero cinco preguntas ,donde los datos random van del uno al diez y con las “a” y “b” y con la función fraction se hace formato fracción, el segundo nivel de suma es con datos más complejos. Si tienes diez correctas te damos la opción de seguir jugando la segunda parte del nivel. Donde funciona igual que las suma, nada más cambia el resultado que en lugar de sumar se multiplica. Si tienes 15 correctas pasamos al segundo nivel de multiplicación con datos más complejos. Si tienes 20 correctas completas el nivel y te imprime felicidades, pero si no tienes las 20 bien, te imprime que no lo completaste y te imprime cuantos correctas de 20 tienes bien.

Después, sigue la suma de matrices donde la función de la matriz tiene dos renglones y tres columnas, donde los datos son random y tenemos otra función con otra matriz que hace lo mismo para poder hacer otra matriz con diferentes datos. La tercera matriz es para que el ordenador sepa cual es el resultado. La cuarta matriz es donde el usuario pone el resultado que él crea que sea de las primeras dos. Si la cuarta matriz es igual a la tercera entonces está bien. El acumulador correcto es más uno, si no se resta e incorrectas es más uno, hay un time sleep para poder separar la impresión de las primeras matrices con el if si está correcto o no. Si tienes 10 correctas completas el nivel y si no te imprime cuantas tienes incorrectas y cuantas correctas.

Sigue el juego de Tic Tac Toe, donde nos imprime el bienvenido con tu nombre, después está un delay, sigue la pregunta si quieres X o O, si escoges X la computadora será O o viceversa. Después, tenemos la función para crear el tablero, que son puros print con punto format para que se imprima la casilla donde quieres que vaya tu ficha. Después, viene otra función para ver qué pasa si quedamos empatados, si se ocupan todas las casillas y nada junta tres, es un empate. Sigue, la función de victoria donde declaramos las posibilidades o las opciones de como puedes ganar. Después, tenemos la función del movimiento del

jugador donde tenemos que seleccionar la casilla que queremos, si la casilla no está dentro de 0 a 8, te imprime seleccione una opción variable. Sigue la función del movimiento del ordenador, que con un for va checar casilla por casilla para poder poner su ficha y tratar de ganar al jugador. Después sigue un while donde se hace un bucle donde llama las funciones de que se necesitan para jugar y se rompe cuando uno gana o pierde o se empata.

Proyecto Final del Curso:

Descripción Entrega del Proyecto:

Lineamientos para entrega final:

Se debe considerar entregar antes del inicio de la clase de exposición:

- Código fuente
- Documento descriptivo de la solución desarrollada.

Código fuente

Código fuente funcional y libre de errores, en archivo .py (No se admitirá otro tipo de formato, proyecto o lenguaje.). Debe ser la misma versión expuesta ante el grupo.

Código fuente comentado, con las siguientes consideraciones:

- Identificación y breve explicación de principales secciones (menús, juegos, resultados, estadística)
- Identificación y breve explicación de las funciones utilizadas
- Identificación y breve explicación de estructuras de datos
- Identificación y breve explicación de manejo de archivos.
- Identificación y explicación de tema nuevo investigado

Documento descriptivo de la solución desarrollada.

Documento descriptivo de la solución desarrollada, el cual especifique:

Descripción detallada de la solución (versión final, de la presentada en la primera etapa) Inventario de funciones (con paso y sin paso de parámetro, con y sin retorno)

Inventario de matrices

Inventario de listas

Inventario de funciones y librerías vistas utilizadas

Inventario de funciones y librerías investigadas

Funciones con paso de parámetro con retorno	Funciones sin paso de parámetro con retorno	Funciones con paso de parámetro sin retorno	Funciones sin paso de parámetro sin retorno	Matrices	Listas	Inventario de ciclos usados	Funciones especiales y librerías	Funciones de string usadas	Tipos de acceso a archivos	Funciones y librerías investigadas
def suma_us():			def main()	mat3	list a=[]	if	from fractions import Fraction			from fractions import Fraction
def matriz_random():			def main_suma_matrices(nombre):	matriz 1=	list a=[]	for	import random			import os
def suma_matrices(matriz1,matriz2):			def suma_matrices(nombre):	matriz 2=	posiciones =[0 ,1,2 ,3,4 ,5,6 ,7,8]	while	import time			
def juego_gato(nombre):			def potencias(nombre):	matriz 4	posiciones =[0 ,1,2 ,3,4 ,5,6 ,7,8]	elif	import pandas as pd			
def empate(matriz):			def tablero():	matriz =[" "]*9		else	import os			
def victoria(matriz):			def movimiento_jugador()							
			def movimiento_ordenador():							

Conclusiones:

Con este proyecto pude aprender la importancia de saber las bases de las matemáticas en primaria, ya que estas bases se siguen ocupando para los grados más adelante. Por ejemplo, yo no sabía muy bien sumar fracciones o restas fracciones, por eso implementé la suma de fracciones. Yo creo que actualmente los padres de familia no se encargan de vigilar y cuidar de que los niños hagan su tarea y hoy en día a muchas aplicaciones que existen en internet y tú pones el problema y te dan el resultado sin saber cómo llegaron a tal resultado. Fue un reto hacer el gato, el crear el tablero, decir a la computadora que también jugara, pero me sirvió para aprender cosas nuevas.