

Instituto Tecnológico de Costa Rica

Ingeniería en Computación

Taller de programación

Sudoku

Estudiante: Pía Sancho Zamora

Profesor: William Mata

Fecha: Noviembre 2019

Tabla de contenido:

Enunciado del proyecto.....	p3
Temas investigados.....	p4
Diseño de la solución.....	p6
Conclusiones.....	p7
Estadísticas de trabajo.....	p8
Rúbrica.....	p9
Manual de usuario.....	p11

Enunciado del proyecto

El sudoku es un pasatiempo de periódicos que consiste en llenar con dígitos del 1 al 9 cada una de las casillas de una cuadrícula que tiene un total de 9 x 9 casillas. La cuadrícula la podemos ver como una matriz de 9 filas y 9 columnas. A la vez esta cuadrícula se divide en subcuadrículas de 3 x 3 casillas. Se toman como base algunos dígitos fijos en la cuadrícula, lo cual determina el nivel de dificultad para completar el juego. Aunque usualmente se usan dígitos, lo que interesa es que sean grupos de 9 elementos diferentes: por ejemplo 9 letras, 9 colores, 9 frutas, etc. La regla de este juego es que un mismo elemento no se puede repetir: - En una misma fila (casillas horizontales) - En una misma columna (casillas verticales) o en el mismo cuadrante de 3x3 alrededor del número.

El objetivo del proyecto era replicar un sudoku en Python. El programa tendrá un menú principal desde el cual se accederá la funcionalidad del programa, es decir, lo que el programa hace. Usted puede agregar otras funcionalidades que vayan a mejorar el producto. En la interfaz gráfica ponga atención a los diferentes elementos como son los tamaños de letras, colores, formas, menús, botones, cuadros de texto, etc. Puede hacer cambios a la interfaz siempre y cuando cumpla con los requerimientos del programa que se indican seguidamente.

El programa va a tener botones de inicio del juego, cargar partida, ayuda, acerca de, configurar, partidas y salir. El juego contara con un top 10 y con botones para seleccionar el número que se quiere poner.

Con este proyecto se quiere reforzar los conocimientos acerca de la metodología de desarrollo de programas de mayor escala. Entender el problema. Diseñar la solución: dividir el problema en problemas más pequeños. Codificar la solución. o Probar y evaluar el programa. Aplicar y reforzar aspectos del lenguaje Python 3 o Uso de estructuras condicionales y de repetición de procesos. Desarrollo de funciones. Utilización de archivos. Utilización de estructuras de datos nativas de Python y TDA (Tipos de Datos Abstractos). Uso de clases (POO) en la funcionalidad de los jugadores del Top-10. Ampliar el conocimiento acerca del desarrollo de GUI en Python. Aplicar buenas prácticas de programación: documentación interna y externa del programa, reutilización de código, nombres significativos, eficiencia del programa, evaluar alternativas, uso de técnica divide y vencerás (dividir el problema en partes, desarrollar cada una de esas partes), etc. Usar algún software de control de versiones de software, por ejemplo Git o algún otro que usted decida. En los próximos días se dará un taller sobre este tipo de software. - Validación de los datos de entrada: todos los datos de entrada se deben validar según restricciones que se indican en cada uno de ellos y fomentar en el estudiante la investigación

Temas investigados:

Para este proyecto se tuvieron que investigar varios procesos y modos que no se vieron en clase para poder concluir el proyecto.

Se tuvieron que utilizar las siguientes librerías de Python:

- Tkinter

Se utilizó para el funcionamiento total de la interfaz.

- Pickle

Se utilizó para guardar archivos, es una librería popular usada para esto mismo

- Os

Se utilize para los archivos, para poder guardar las partidas, se puede usar en conjunto de pickle y dump para los archivos

- Webbrowser

Se utilizó para poder abrir el archivo pdf de la documentación

- Messagebox

Se utilizó para poder desplegar varios mensajes en una especie de caja de mensajes, se puede dar solo como aviso, advertencia o preguntas de sí o no.

- Pygame

Se utilizó pygame para poder ponerle música al programa.

- Random

Se utilizó para obtener números random y escoger matrices

Se tenía un poco de conocimiento previo de Tkinter pero se tuvo que investigar estas librerías y entre otros aspectos, como la utilización de botones, como Radiobuttons, que son botones circulares, o se pueden usar botones diferentes que tienen apariencia de barra. Colores, los cuales para utilizarlos hay que asignar un número a cada uno, música, nuevas ventanas, messageboxes como previamente mencionado y toda la interfaz en sí.

También se tuvo que investigar sobre como usar las funciones de los archivos que aunque fueron vistas en clase se tuvieron que reforzar ya que son confusas de usar, como las de read, open, write, close, etc.

Todo esta hecho funcionalmente, a punta de listas, matrices, for y whiles que fueron los vistos en clase.

Se utilizaron clases para hacer el grid de los botones del juego en sí.

Se utlizaron diccionarios para guardar la información de los colores de los números y para guardar las matrices de niveles fácil, medio y difícil.

Los diccionarios son un tipo de estructura de datos con características especiales que nos permiten almacenar cualquier tipo de valor o hasta funciones.

Diseño de la solución:

Se empezó creando matrices de juegos ya existentes en internet con diferentes niveles.

Para guardar cada matriz se utilizaron diccionarios.

El usuario tiene la función de colocar la información que quiere en los entrys. Cada vez que el usuario coloque un entry, será el nombre del jugador y será agregado al top 10.

Se tuvieron que hacer funciones para validar cuadrantes, filas y columnas, para la función de cuadrantes se tuvo que realizar varias veces porque no brindaba los cuadrantes bien.

La matriz se actualiza cada vez que se coloca un número y una posición de adonde se quiera colocar el número, se tuvieron que hacer varias funciones para conseguir la posición de un botón porque se presentaron varios problemas donde la posición no estaba bien y para solucionarlo se tuvieron que recurrir a hacer funciones con clases.

Otro aspecto que dio problemas fue el timer, para el cual se tuvo que hacer la librería threading porque si no se hacía imposible.

La librería de tkinter para música winsound no funcionaba entonces se tuvo que recurrir a otra librería originaria de Python llamada pygame en la cual insertarle música al programa se hace mucho más fácil.

Conclusiones

Problemas encontrados y soluciones a las mismas

- Manejo de archivos. A la hora de manejar archivos se complicó un poco porque no se guardaba la información de la matriz una vez cerrado el shell. Para solucionarlo se usaron las funciones read, open y write en un txt.
- Manejo de interfaz en sí. El lograr conectar el grid de los botones con la ventana del juego. La cantidad de variables puede ser muy confusa, por eso para solucionarlo se separaron todas las variables, botones, etc adentro de funciones y poniendo casi que todas las variables globales para poder usarlas.
- Se reforzó el uso de diccionarios ya que se pudo guardar la información de cada matriz y como va cambiando conforme a como se le van arreglando los botones.
- Se reforzó la investigación de nuevas librerías que no son tan conocidas y sus usos ya que son tan útiles en diferentes tareas de Python.
- Se hizo mucho más fácil trabajar con más facilidad por los conocimientos adquiridos anteriormente de los últimos dos proyectos.

Estadística de tiempos:

Actividad Realizada	Horas
Análisis del problema	3 horas
Diseño de algoritmos	12 horas
Investigación de programación	4 horas
Documentación interna	1 hora
Pruebas	2 horas
Elaboración del manual de usuario	2 hora
Elaboración de documentación	2 horas
Total	26 horas

Rúbrica de evaluación y otros aspectos

Concepto	Puntos	Puntos obtenidos	Avance 100/0	Análisis De resultados
Menú	1		100	
Opción Jugar	15		100	
Botón Iniciar Juego	10		100	
Botón borrar jugada	5		50	No borra la jugada si no que se sale del programa en sí
Botón terminar juego	2		100	
Botón borrar juego	5		100	
Botón Top 10	10		100	
Botón guardar juego	5		100	
Botón cargar juego	15		100	
Opción configurar	10		50	No configura el timer el timer siempre va a existir
Ayuda	10		100	
Acerca de	1		100	
Salir	1		100	
Reloj y timer	10		50	Solo tiene timer no reloj

Funcionalidades adicionales	Se incluyó música y un botón para ver la solución		100	
Total	100			

Manual de usuario

Manual que explica al usuario como usar el programa

file:///C:/Users/user/Documents/programacion%20William/programa3_piasancho/sudoku/manual_de_usuario_sudoku.pdf

El programa desplegará un menú con un botón de jugar partida, en el que se encuentra el sudoku, un espacio para poner su nombre que lo guardara en un top 10, un botón para poder ver la solución, botones con los números que se puedan poner en el tablero del sudoku. Botón para guardar la partida.

El sudoku es un juego de periódicos que se llena con números en casillas de una tabla de 9x9. Los números van del 1 al 9, pero no se pueden repetir en la misma fila, columna o cuadrante. Cuadrantes son las casillas que estén alrededor de la posición seleccionada. Son de 3x3.

El programa cuenta con otros botones, por ejemplo un timer, configuración para cambiar los niveles, se puede jugar bajo tres diferentes niveles, fácil, medio y difícil. Entre más difícil sea el nivel más espacios vacíos van a haber.

También cuenta con un botón de top 10 donde su nombre entrará al top si está entre los mejores tiempos.

Cuenta con un botón para guardar partidas en caso de que guardar la partida y jugar después.

Se puede configurar el timer, si el timer se acaba, se acaba la partida o si se llenan todas las casillas. En el caso de que sean correctas se gana, si no están correctas puede continuar jugando o ver la solución con un botón.

El programa cuenta con música

El botón ayuda desplegará este manual de usuario de ayuda de como utilizar el programa.

El botón acerca de desplegará información sobre la versión del programa y el autor, etc.

El botón salir, sale del programa.

			1			7	4	
	5			9			3	2
		6	7			9		
4			8					
	2						1	
					9			5
		4			7	3		
7	3			2			6	
		5			4			

