Proyecto de Programación – Sudoku Melo

Integrantes:

- Santiago Jaimes

- Alejandra Pardo

- Sebastián Bernal

Resumen Ejecutivo:

Los Sudokus son pasatiempos que pueden llegar a ser bastante interesantes, entretenidos y a veces desafiantes, sin embargo, un problema identificado es lo tedioso y frustrante que puede llegar a ser enfrentarse a un juego de Sudoku muy complejo o difícil, o de no saber si la solución a un puzzle, por ejemplo, del periódico es la correcta, por lo que surge la necesidad de un programa para ayudar a encontrar soluciones a estos juegos de Sudoku. La solución al problema es la creación de un programa en Python el cual, de la solución correcta a cualquier Sudoku, sin importar su dificultad, comprobar si tiene solución, o comprobar si una solución planteada por el usuario es correcta.

Funcionalidad de la Herramienta:

El producto tiene tres funciones principales; La de jugar, en la cual el programa genera un Sudoku con tres niveles de dificultad para que el usuario lo resuelva, con la capacidad de verificar su solución o en caso de no poder terminarlo dejar que el programa lo resuelva, también se encuentra la función de verificar en la cual el usuario puede ingresar un Sudoku junto con su solución para verificar si es la correcta o si ha cometido algún error y en la última funcionalidad el usuario puede ingresar un Sudoku planteado para que el programa lo resuelva, e incluso responder si el Sudoku propuesto es válido. Como se ve estas herramientas permiten la solución del problema al poder permitirle a una persona solucionar un Sudoku en el cual puede estar estancado o verificar si su manera de completar un puzzle fue la correcta, de igual manera al permitir al usuario jugar y ver la solución si no le fue posible completar el Sudoku generado.

Repositorio GitHub: <https://github.com/Santiago1325/Sudoku->

Descripción de la herramienta:

La construcción del producto esta “dividida” en tres grandes partes. La primera es la generación de un Sudoku, en esta sección se incluyeron las funciones que crean un Sudoku aleatorio para ser resuelto por el usuario. El Sudoku se puede definir como una “lista de listas”, donde cada fila es representada por una sub-lista dentro de una lista y cada columna como las posiciones de esas sub-listas. Se comienza partiendo de un juego de Sudoku completo y valido, en un Sudoku es posible hacer permutaciones para generar cada vez un juego diferente, estas permutaciones tienen ciertas limitaciones para evitar que al intercambiar ciertos elementos se pierda la validez del Sudoku pero que sin embargo garantizan la aleatoriedad del Sudoku, estas limitaciones son: Solo se puede intercambiar filas y columnas en grupos de a tres, es decir es posible intercambiar entre la fila o columna uno, con las numero dos y número tres, pero no es posible con alguna otra ya que podría causar que el Sudoku deje de ser válido, de igual forma se puede intercambiar las filas y columnas de sub-cuadriculas de tres por tres, pero no se puede individualmente entre ellas. Al hacer estas permutaciones una cantidad aleatoria de veces (con ayuda del módulo “random”) se puede conseguir un Sudoku diferente cada vez. Hay tres niveles de dificultad para escoger, siendo tres el más difícil, entre más dificultad se tiene menos pistas, por lo que dependiendo de la dificultad seleccionada se remueven más o menos números de forma aleatoria, teniendo en cuenta que el mínimo de pistas para que un Sudoku sea válido es de diecisiete. Adicionalmente esta sección incluye funciones que retornan la fila, columna o sub-cuadricula en la que se encuentra cierto número e incluso es posible generar un Sudoku vacío, estas funciones son útiles para las siguientes partes.

La segunda parte se encarga de encontrar la solución de un Sudoku, lo que hizo que esto fuera posible es la *recursión,* que es definido como la forma en la cual se especifica un proceso basado en su propia definición, aplicando esto en el proyecto, se concibe como una función que se llama a sí misma. Una de estas funciones es conocida como “Backtracking”, cuyo funcionamiento se basa en ir probando todas las soluciones posibles hasta encontrarse en una posición en la cual no es posible avanzar, por lo que deberá volver atrás y seguir por un diferente camino. En el caso de este proyecto, el algoritmo busca la primera casilla vacía, en la cual empieza a probar con un valor inicial y todas las posibles soluciones de ese valor, repitiendo el mismo proceso con todas las casillas vacías hasta dar con la solución final, con ayuda de las funciones definidas en la parte anterior el algoritmo, antes de poner un número, verifica si no rompe las reglas del Sudoku al ponerlo (No es “legal” repetir un numero en una fila, columna o sub-cuadricula), así determina que valores son los que puede utilizar para encontrar la solución. Al igual que el modulo anterior, este incluye funciones que serán útiles en la última parte, por ejemplo las que verifican la validez y completitud de un Sudoku.

En la parte final, con la ayuda del módulo Pygame, se realizó lo que es la interfaz gráfica del producto, en la cual se puede navegar por las distintas funciones principales del producto como lo son, generar un Sudoku para jugar, verificar una solución planteada, o ingresar un Sudoku planteado para ver su solución, aquí es donde las funciones del módulo anterior para verificar empiezan su funcionamiento, ya que con estas funciones y con las del primer módulo se hace la verificación de cada uno de los valores del Sudoku para validar una solución o un Sudoku sin resolver, más que todo incluye funciones que hacen posible el funcionamiento del producto con una interfaz gráfica ya que permiten que sea mostrado todo el contenido en la pantalla y que sea fácil de interactuar para el usuario