

Proyecto Programado #3 Sudoku

Introducción

Sudoku (en japonés: 数独, sūdoku) es un juego matemático que se publicó por primera vez a finales de la década de 1970 y se popularizó en Japón en 1986, dándose a conocer en el ámbito internacional en 2005 cuando numerosos periódicos empezaron a publicarlo en su sección de pasatiempos. El objetivo del sudoku es rellenar una cuadrícula de 9×9 celdas (81 casillas) dividida en subcuadrículas de 3×3 (también llamadas "cajas" o "regiones") con las cifras del 1 al 9 partiendo de algunos números ya dispuestos en algunas de las celdas. Aunque se podrían usar colores, letras, figuras, se conviene en usar números para mayor claridad, lo que importa, es que sean nueve elementos diferenciados, que no se deben repetir en una misma fila, columna o subcuadrícula. Un sudoku está bien planteado si la solución es única, algo que el matemático Gary McGuire ha demostrado que no es posible si no hay un mínimo de 17 cifras de pista al principio.

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

¿Qué se busca con este proyecto?

El objetivo general de este proyecto es facilitar un acercamiento con el juego de Sudoku desde la perspectiva de la programación lógica, de manera que las partes del juego y las reglas que lo enmarcan sean diseccionadas minuciosamente para desarrollar una base de conocimiento que permita disfrutar de este juego por medio de Prolog.

1. Practicar las habilidades de desarrollo de software.
2. Ejercitar la toma de decisiones sobre el dominio del problema y de la solución.
3. Aplicar los conceptos del paradigma lógico.

Software a desarrollar

Este proyecto requiere de un conocimiento detallado de las reglas del juego de Sudoku. Antes de iniciar el trabajo debe comprender por completo el funcionamiento del juego, ya que lo que se desea es reproducir el comportamiento de este.

Asegúrese de conocer a la perfección las reglas del juego, incluyendo sus variantes. Analice la composición del juego.

El juego debe disponer de las siguientes funcionalidades:

Tablero:

Se deberá disponer de un tablero de dimensiones 9 x 9 donde se permitirá realizar el juego y donde se mostrará el Sudoku. Le permite al usuario indicar o escribir un número en las posiciones de ingreso de dígitos (ver ejemplos de juegos en línea de Sudoku). Puede también indicar por otro medio por ejemplo indicando posición según eje X y Y por medio de un botón.

Generar nuevo juego:

Debe generar un nuevo juego de Sudoku, se debe recargar el tablero con el nuevo Sudoku en su estado inicial (las celdas de ingreso de dígitos limpias). Debe considerar que no se permiten cadenas de soluciones de un solo dígito. Se debe generar un nuevo Sudoku 100% aleatorio, tomando en cuenta las reglas del juego para generarlo. Deben existir cadenas de solución que se combinan celdas en su solución (una celda funciona para una cadena vertical y para una horizontal). La generación debe ser 100% aleatoria utilizando Prolog, indicando al menos 17 pistas, máximo 25 pistas.

Reiniciar:

Permite regresar el Sudoku a su estado inicial.

Verificar:

Verifica cómo se está resolviendo el Sudoku, indicando la cantidad de errores si los hay y la cantidad de campos o dígitos pendientes por indicar, por ejemplo: “hay dos dígitos incorrectos y hay 41 celdas vacías de 64”, “no hay dígitos incorrectos pero hay 41 celdas vacías de 64” o “juego finalizado exitosamente, felicidades!!!” según el estado que tenga el Sudoku en ese instante.

Sugerencia:

Coloca un número de forma aleatoria en una casilla. Entre paréntesis se muestran la cantidad de sugerencias disponibles. Se inicia con 5 sugerencias permitidas.

Ver solución:

Debe mostrar en el tablero el Sudoku resuelto indicando el número que corresponde en cada celda de ingreso de dígitos.

Estadísticas:

Por cada Sudoku jugado en la sesión (lo que dura el sistema abierto) se debe indicar lo siguiente:

- a) Cantidad de celdas de ingreso de dígitos
- b) Cantidad de verificaciones realizadas
- c) Cantidad de errores de verificación
- d) Cantidad de sugerencias utilizadas
- e) Tipo Finalización: exitosa, abandono (por nuevo juego), autosolución.

.

Opcionales

Las siguientes características no son obligatorias, pero se asignarán puntos extras en caso de que se desarrollen. Sólo se otorgarán puntos extra si las funcionalidades descritas anteriormente están completas.

- Guardar repetición. Permitir al usuario la opción para guardar y reproducir una repetición en un archivo en disco duro. 5ptos
- Redimensionar: Permita jugar el Sudoku 16x16. 10 ptos
- Implementar cronómetro para registrar el tiempo de juego de cada partida (Esto aplica si se cumple con todos los requerimientos del juego y se debe incluir en estadísticas). 5 ptos
- Se darán **2.5 puntos adicionales** al entregar a más tardar el miércoles 20 de mayo a las 11:55:55 PM el Documento de Requerimientos, ver plantilla suministrada en el Tec Digital. Debe subirse en la documentación llamada "Proyecto Programado 3 (archivos adicionales)" debajo de la carpeta de "Proyectos".

Aspectos técnicos

La generación del juego (**Generar nuevo juego**) 100% aleatoria, sugerencia y la verificación debe realizarse en SWI Prolog y hacer una liga con Java o C# (opcional Web) para realizar la restante funcionalidad, la lógica del juego debe generarse en Prolog. En caso de requerir librerías adicionales para compilar y ejecutar el programa, deberán especificarlo en la documentación, ya que de lo contrario se descontarán puntos en la evaluación.

Deberán utilizar el sistema de control de versiones GitHub, el repositorio deberá ser público o incluir al profesor en el control de acceso del mismo. Se utiliza para evaluar la correcta gestión del tiempo y trabajo colaborativo.

Documentación

La documentación es un aspecto de gran importancia en el desarrollo de programas, especialmente en tareas relacionadas con el mantenimiento de estos.

Para la documentación interna, deberán incluir comentarios descriptivos para cada función, con sus entradas, salidas, restricciones y **objetivo**.

La documentación externa deberá incluir:

1. Portada.
2. Manual de usuario: **instrucciones de compilación, ejecución y uso**.
3. Pruebas de funcionalidad: incluir *screenshots*.
4. Descripción del problema.
5. Diseño del programa: decisiones de diseño, algoritmos usados.
6. Librerías usadas: manejo entradas-salidas, archivos, etc.
7. Análisis de resultados: objetivos alcanzados, objetivos no alcanzados, y razones por las cuales no se alcanzaron los objetivos (en caso de haberlos).
8. Bitácora (autogenerada en git, commit por usuario incluyendo comentario)

Forma de trabajo

El trabajo se debe realizar en parejas.

Evaluación

La evaluación se va a centrar en dos elementos: programación y documentación.

La tarea tiene un valor de **12.5%** de la nota final, en el rubro de Proyectos.

Desglose de la evaluación de la tarea programada:

1. Documentación interna 2 puntos.
2. Documentación externa 8 puntos.
3. Funcionalidad 80 puntos.
4. Revisión del proyecto (según completitud del proyecto y gestión del tiempo) 5 puntos.
5. Hora de Entrega 5 puntos.

Aspectos administrativos

Debe crear un archivo **.zip** ("PP3_Integrante1_Integrante2.zip") que contenga únicamente un archivo **info.txt** y 2 carpetas llamadas **documentacion** y **programa**, en la primera deberá incluir el documento de *word* o pdf solicitado y en la segunda los archivos y carpetas necesarias para la implementación de este proyecto programado, y/o link en git del repositorio. El archivo **info.txt** debe contener la siguiente información (cualidades):

- a. Nombre del curso
- b. Número de semestre y año lectivo
- c. Nombre de los Estudiantes
- d. Número de carnet de los estudiantes
- e. Número del proyecto programado
- f. Fecha de entrega
- g. Estatus de la entrega (debe ser **CONGRUENTE** con la solución entregada):
[Deplorable | Regular | Buena | MuyBuena | Excelente | Superior]

Entrega

Deberá subir el archivo antes mencionado al TEC Digital en el curso de LENGUAJES DE PROGRAMACIÓN GR 60, en la asignación llamada “P3” debajo del rubro de “Proyectos”. En la evaluación del Proyecto el rubro de “Hora de Entrega” valdrá por 5 puntos de la nota total del proyecto, según la siguiente escala:

- a. Si se entrega antes de las 11:55:55 **PM** del lunes 02 de junio de 2025, 5 puntos.
- b. Si se entrega antes de las 11:55:55 **AM** del martes 03 de junio de 2025, 2.5 puntos.
- c. Si se entrega antes de las 11:55:55 **PM** del martes 03 de junio de 2025, 0 puntos.

NO SE ACEPTARÁN trabajos que contengan “commits” posterior a esta fecha.

Todo el contenido de cada proyecto debe ser 100% original y en caso de plagio todos los integrantes del grupo tendrán nota cero.

Todos los miembros del grupo deberán participar de la revisión donde se explique la funcionalidad y se demuestra la autoría del proyecto.