

Universidad Rafael Landívar.

Facultad de Ingeniería. Carrera de INGENIERÍA INDUSTRIAL.

Pensamiento computacional

Docente: Ovalle Arrecis Luis Pedro

Proyecto 2

“AJEDREZ”

Estudiante: Sáenz Bojórquez Denisse Mishell.

Carnet: 1269324.

Guatemala, día 22 de mayo de 2024

¿Qué acciones debe poder hacer su programa? Enumérelas.

1. Validar el ingreso de una pieza de ajedrez, solicitando al usuario el tipo de pieza, el color y la posición en el tablero, y asegurándose de que la entrada sea válida.
2. Imprimir el tablero de ajedrez actual, mostrando las posiciones de las piezas.
3. Calcular los movimientos válidos hacia arriba desde una posición dada en el tablero.
4. Calcular los movimientos válidos hacia abajo desde una posición dada en el tablero.
5. Calcular los movimientos válidos hacia la izquierda desde una posición dada en el tablero.
6. Calcular los movimientos válidos hacia la derecha desde una posición dada en el tablero.
7. Listar todos los movimientos válidos de una pieza en particular en el tablero, considerando su posición actual y el estado del tablero.

¿Con qué datos va a trabajar? ¿Qué información debe pedir al usuario?, defina sus datos de entrada.

1. Tipo de pieza: El usuario debe ingresar el tipo de pieza que desea colocar en el tablero. Esto puede ser una torre, alfil, peón, caballo, reina o rey.
2. Color de la pieza: El usuario debe especificar el color de la pieza, que puede ser blanco o negro.
3. Posición en el tablero: El usuario debe proporcionar la posición donde desea colocar la pieza. Esto se hace especificando la fila (letra de la A a la H) y la columna (número del 1 al 8) en el tablero.

Por lo tanto, los datos de entrada son:

- Tipo de pieza (cadena de texto).
- Color de la pieza (cadena de texto).
- Fila de la pieza (letra de la A a la H).
- Columna de la pieza (número del 1 al 8).

Estos datos se utilizarán para validar la entrada del usuario y para representar la pieza en el tablero, así como para calcular los movimientos válidos de la pieza ingresada.

¿Qué variables utilizará para almacenar la información? Defina el nombre y el tipo de dato que utilizará para gestionar estos datos principales

1. **tipo_pieza:** Una cadena de texto que almacenará el tipo de pieza ingresado por el usuario (torre, alfil, peón, caballo, reina o rey).
 - Tipo de dato: String (str).
2. **color:** Una cadena de texto que almacenará el color de la pieza ingresado por el usuario (blanco o negro).
 - Tipo de dato: String (str).
3. **fila_letra:** Una cadena de texto que almacenará la fila en la que el usuario desea colocar la pieza, representada por una letra de la A a la H.
 - Tipo de dato: String (str).
4. **columna_numero:** Un entero que almacenará la columna en la que el usuario desea colocar la pieza, representada por un número del 1 al 8.
 - Tipo de dato: Entero (int).
5. **tablero:** Una lista bidimensional que representará el tablero de ajedrez. Cada elemento de la lista contendrá una cadena de texto representando el estado de una casilla del tablero.
 - Tipo de dato: Lista de listas de cadenas de texto (List[List[str]]).

¿Qué condiciones o restricciones debe tomar en cuenta? ¿Qué cálculos debe hacer?

1. **Validación del tipo de pieza:** El programa debe asegurarse de que el tipo de pieza ingresado por el usuario sea uno de los tipos válidos en ajedrez (torre, alfil, peón, caballo, reina o rey).
2. **Validación del color de la pieza:** El programa debe verificar que el color ingresado por el usuario sea blanco o negro, los únicos colores válidos en el ajedrez.
3. **Validación de la posición en el tablero:** El programa debe comprobar que la posición ingresada por el usuario esté dentro de los límites del tablero de ajedrez (filas de la A a la H y columnas del 1 al 8).
4. **Verificación de ocupación de casilla:** Antes de colocar una nueva pieza en el tablero, el programa debe verificar que la casilla seleccionada esté vacía para evitar la colocación de dos piezas en la misma posición.
5. **Cálculo de movimientos válidos:** Para cada tipo de pieza, el programa debe calcular los movimientos válidos que la pieza puede realizar desde su posición actual. Esto implica considerar las reglas específicas de movimiento de cada tipo de pieza (por ejemplo, la torre se mueve horizontal o verticalmente, el alfil diagonalmente, etc.).

6. Actualización del tablero: Después de que el usuario haya ingresado una nueva pieza, el programa debe actualizar el estado del tablero colocando la pieza en la posición especificada por el usuario.
7. Impresión del tablero actualizado: El programa debe imprimir el tablero de ajedrez actualizado después de que se haya colocado una nueva pieza, para que el usuario pueda ver la ubicación de todas las piezas en el tablero.

Al considerar estas condiciones y restricciones, el programa garantizará que el juego de ajedrez se desarrolle de acuerdo con las reglas establecidas y que las acciones del usuario sean válidas y coherentes con el estado del juego.

¿Qué funciones implementará?

1. `validar_ingreso_pieza(tablero)`: Esta función solicitará al usuario el tipo de pieza, el color y la posición en el tablero donde desea colocar la pieza. Luego, validará la entrada del usuario para garantizar que sea correcta y devolverá los datos ingresados.
2. `imprimir_tablero(tablero)`: Esta función imprimirá en la consola el estado actual del tablero de ajedrez, mostrando la posición de las piezas.
3. Funciones para calcular movimientos válidos:
 - `movimientos_arriba(tablero, fila, columna)`: Calcula los movimientos válidos hacia arriba desde una posición dada en el tablero.
 - `movimientos_abajo(tablero, fila, columna)`: Calcula los movimientos válidos hacia abajo desde una posición dada en el tablero.
 - `movimientos_izquierda(tablero, fila, columna)`: Calcula los movimientos válidos hacia la izquierda desde una posición dada en el tablero.
 - `movimientos_derecha(tablero, fila, columna)`: Calcula los movimientos válidos hacia la derecha desde una posición dada en el tablero. Estas funciones serán utilizadas para calcular los movimientos válidos de diferentes tipos de piezas en el tablero.
4. `listar_movimientos_validos(tablero, fila, columna)`: Esta función listará todos los movimientos válidos de una pieza en particular en el tablero, considerando su posición actual y el estado del tablero.

Estas funciones trabajarán en conjunto para permitir al usuario ingresar piezas en el tablero, calcular los movimientos válidos de estas piezas y mostrar el estado actualizado del tablero después de cada jugada.

Algoritmo que implementará en el programa, descrito mediante el Diagrama de Flujo elaborado en Draw.io, para mostrar la lógica de las diferentes acciones. Realizar un diagrama de flujo por cada función y un diagrama general del programa principal.

