Universidad Rafael Landívar.

Facultad de Ingeniería.

Ingeniería Civil.

Laboratorio de Pensamiento Computacional

**Docente**: Ing. Luis Ovalle

**PROYECTO DE LABORATORIO No.2**

**Estudiante**:

Kreitz Caravantes, Juan Esteban

**Carné**: 1198024

Guatemala, 22 de mayo de 2024

ÍNDICE

[¿QUÉ ACCIONES DEBE PODER HACER EL PROGRAMA? 3](#_Toc167007853)

[¿CON QUÉ DATOS SE TRABAJARAN? Y ¿QUÉ INFORMACIÓN SE LE PEDIRÁ AL USUARIO? 3](#_Toc167007854)

[VARIABLES UTILIZADAS 4](#_Toc167007855)

[CONDICIONES O RESTRICCIONES A TOMAR EN CUENTA, CÁLCULOS REALIZADOS 4](#_Toc167007856)

[FUNCIONES IMPLEMENTADAS 5](#_Toc167007857)

[DIAGRAMA DE FLUJO 5](#_Toc167007858)

¿QUÉ ACCIONES DEBE PODER HACER EL PROGRAMA?

El programa realizará las siguientes acciones:

* Inicializar el tablero vacío.
* Pedir al usuario el ingreso de piezas (nombre, color, posición) hasta que el usuario escriba "fin".
* Verificar que las posiciones ingresadas para las piezas no estén ocupadas.
* Colocar las piezas ingresadas en las posiciones correspondientes del tablero.
* Pedir al usuario el ingreso de una torre (color, posición).
* Verificar que la posición ingresada para la torre no esté ocupada.
* Colocar la torre en la posición correspondiente del tablero.
* Imprimir el estado del tablero después de cada ingreso de pieza.
* Determinar y mostrar los movimientos posibles de la torre en las cuatro direcciones (derecha, izquierda, arriba, abajo).
* Verificar si las casillas en las que se mueve la torre están vacías o contienen piezas del color opuesto.
* Listar los posibles movimientos de la torre con sus coordenadas.

¿CON QUÉ DATOS SE TRABAJARAN? Y ¿QUÉ INFORMACIÓN SE LE PEDIRÁ AL USUARIO?

Los datos que se le pedirán al usuario serán los siguientes:

* **Nombre de la pieza**: en un string
* **Color de la pieza**: en un string
* **Columna de la pieza**: en un character (de 'a' a 'h').
* **Fila de la pieza**: en un int (de 1 a 8)
* **Color de la torre**: en un string
* **Columna de la torre**: en un character (de 'a' a 'h')
* **Fila de la torre**: en un integer (de 1 a 8)

VARIABLES UTILIZADAS

Las variables que se utlizaron fueron:

1. **Tablero**: Matriz principal de 8x8.
2. **nombre\_pieza**: Almacena el nombre de la pieza ingresada por el usuario. Tipo: str
3. **color**: Almacena el color de la pieza ingresada por el usuario. Tipo: str
4. **col\_pieza**: Almacena la columna de la pieza ingresada por el usuario. Tipo: str
5. **fila\_pieza**: Almacena la fila de la pieza ingresada por el usuario. Tipo: int
6. **cp**: Almacena la posición de la columna convertida en índice de la matriz. Tipo: int
7. **fp**: Almacena la posición de la fila convertida en índice de la matriz. Tipo: int
8. **color\_torre**: Almacena el color de la torre ingresada por el usuario. Tipo: str
9. **colTorre**: Almacena la columna de la torre ingresada por el usuario. Tipo: str
10. **filaTorre**: Almacena la fila de la torre ingresada por el usuario. Tipo: int
11. **ct**: Almacena la columna de la torre convertida en índice de la matriz. Tipo: int
12. **ft**: Almacena la fila de la torre convertida en índice de la matriz. Tipo: int
13. **colDerechaInicial, colIzquierdaInicial, filSuperiorInicial, filInferiorInicial**: Almacenan las posiciones iniciales para los movimientos de la torre en las diferentes direcciones. Tipo: int
14. **n, q, o, p**: Variables para almacenar la representación de columna y fila en las impresiones de movimientos. Tipo: str para columnas y int para filas.

CONDICIONES O RESTRICCIONES A TOMAR EN CUENTA, CÁLCULOS REALIZADOS

Condiciones:

 Verificar si la casilla donde se quiere colocar una pieza o la torre ya está ocupada.

 Las posiciones ingresadas deben estar dentro de los límites del tablero (filas de 1 a 8 y columnas de 'a' a 'h').

 La torre solo puede moverse horizontalmente o verticalmente hasta encontrarse con el borde del tablero o una pieza.

 La torre solo puede capturar piezas de color diferente.

 Continuar ingresando piezas hasta que el usuario ingrese "fin".

Cálculos realizados:

* **Conversión de la columna de letra a índice de la matriz**.
* **Conversión de la fila de ajedrez a índice de la matriz**.
* **Determinación de posiciones vacías o con piezas en las cuatro direcciones de movimiento de la torre**.
* **Verificación de ocupación y color en las posiciones de movimiento de la torre**.

FUNCIONES IMPLEMENTADAS

Las funciones que se implemetaron fueron:

* **convertir\_columna(columna)**: Convierte una columna de letra (a-h) a su posición correspondiente en la matriz.
* **convertir\_fi(fila)**: Convierte una fila de ajedrez (1-8) a su posición correspondiente en la matriz.
* **Ingreso de piezas**: Un ciclo mientras que pide nombre, color, columna y fila de las piezas hasta que el usuario ingrese "fin".
* **Verificación de ocupación**: Verifica si la casilla donde se desea colocar una pieza o la torre está ocupada.
* **Colocación de piezas**: Inserta la pieza o la torre en la posición correspondiente del tablero.
* **Determinación de movimientos de la torre**: Calcula los movimientos posibles de la torre en las cuatro direcciones y verifica la ocupación y el color de las piezas en esas posiciones.
* **Impresión del estado del tablero**: Imprime el estado actual del tablero después de cada movimiento.

DIAGRAMA DE FLUJO

LINK CARPETA GITHUB