



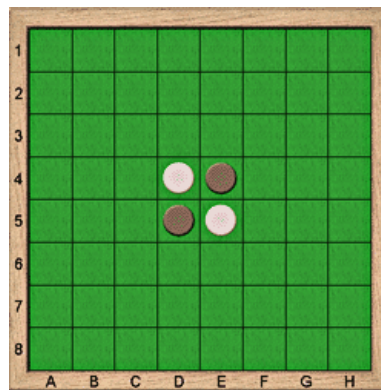
## **PROYECTO I**

Se desea que usted realice en C++ un sistema que simule el conocido juego de Reversi, el cual es un juego entre dos personas, que comparten 64 fichas iguales, de caras distintas, que se van colocando por turnos en un tablero dividido en 64 casillas. Las caras de las fichas se distinguen por su color y cada jugador tiene asignado uno de esos colores, ganando quien tenga más fichas sobre el tablero al finalizar la partida.



El tablero estándar está formado por 8 filas y 8 columnas, y se rige por las siguientes reglas:

- ✓ Al inicio del juego se situarán cuatro fichas (dos de cada color) en las cuatro casillas centrales del tablero, de forma que cada pareja de fichas iguales forme una diagonal entre sí.



- ✓ El jugador que juegue con las negras realiza el primer movimiento, alternando el turno entre uno y otro para los movimientos sucesivos.
- ✓ Los movimientos consisten en incorporar fichas al tablero a razón de una por turno, nunca en desplazar fichas de las que ya estuvieran sobre el tablero.
- ✓ Las incorporaciones deberán hacerse en orden a las siguientes normas:
  - Sólo podrá incorporarse una ficha flanqueando a una o varias fichas contrarias.
  - Por flanquear se entiende el hecho de colocar la nueva ficha en un extremo de una hilera de fichas del color del contrario (una o más fichas) en cuyo extremo opuesto hay una ficha del color de la que se incorpora, sin que existan casillas libres entre ninguna de ellas. Esta hilera puede ser indistintamente vertical, horizontal o diagonal. De este modo,

las fichas del contrincante quedan encerradas entre una que ya estaba en el tablero y la nueva ficha.

- Cada vez que un jugador incorpora una ficha, y por lo tanto encierra a otras del contrario, debe dar la vuelta a las fichas encerradas convirtiéndolas así en propias.
- Si en una sola incorporación se provocase esta situación de flanqueo en más de una línea, se voltearán todas las fichas contrarias que estuvieran implicadas en cada uno de los flanqueos.
- Si no fuera posible para un jugador encerrar a ninguna ficha, deberá pasar en su turno, volviendo el mismo a su oponente.
- Ejemplo de una jugada

	A	B	C	D	E	F	G	H	
1			○	○	○	○	○		1
2				○	●	●	○		2
3	○	○	○	○	○	○	●	○	3
4	○	●	○	●	●	●	↓		4
5	○	●	●	○	○	●	○		5
6	○	○	○	○	●	●	○		6
7			●	●	●	●			7
8			●	●	●	●			8
	A	B	C	D	E	F	G	H	

	A	B	C	D	E	F	G	H	
1			○	○	○	○	○		1
2			○	●	○	○			2
3	○	○	○	○	○	○	○		3
4	○	●	○	○	○	○	○		4
5	○	●	●	○	○	●	○		5
6	○	○	○	○	●	●	○		6
7			●	●	●	●			7
8			●	●	●	●			8
	A	B	C	D	E	F	G	H	

- ✓ La partida finaliza cuando todas las casillas del tablero son ocupadas o ninguno de los 2 jugadores tiene posibilidad de incorporar una nueva ficha. En cualquier caso, vence el jugador que tiene más fichas sobre el tablero.
- ✓ Puede darse el caso de empate si el número de fichas de cada color al acabar la partida es el mismo.

Para el desarrollo del programa, deben tomarse en cuenta las siguientes consideraciones:

- La partida podrá ser realizada de la siguiente manera: humano contra humano, computadora contra humano y computadora contra computadora. La máquina tendrá un nivel de juego en el cual debe tener siempre como meta vencer a su adversario.
- El juego debe ser desarrollado con una interfaz de comandos donde para cada jugada se presente en pantalla el estado del tablero después de la jugada

La entrega debe constar de un informe el cual debe contener al menos: análisis del problema, algoritmo general, estructura de datos (clases), tabla de funciones (asociadas a las clases) y el código fuente del programa (siendo importante el estilo y los comentarios). Analice en su informe la complejidad del algoritmo de jugar de la computadora