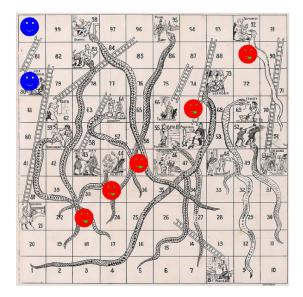
Considere la variación del juego escaleras y serpientes mostrada en la figura:



- La meta del jugador es ganar la partida llegando a una de las casillas marcadas en azul.
- El jugador pierde la partida si cae en una de las casillas marcadas en rojo.
- En cada jugada, *antes de lanzar el dado*, el jugador decide si quiere avanzar o retroceder el número de casillas indicadas por el dado.
- En las casillas 1 y 100 la ficha rebota (si se supera el extremo, se avanza en la otra dirección la cantidad restante).
- 1. Modele este problema como un MDP. De detalladamente todos los elementos del MDP: estados, recompensas, acciones y $p(s', r \mid s, a) \ \forall s, s', r, a$.
- 2. Escriba un módulo de Python para el MDP formulado. Su implementación debe permitir asignar casillas azules y rojas en posiciones arbitrarias del tablero. También debe facilitar la implementación de programación dinámica para hallar la función de valor de una política dada y de iteración de valor para encontrar una política óptima (ver los numerales a continuación).
- 3. Implemente el algoritmo de programación dinámica y use su implementación para encontrar la función de valor de las siguientes políticas, para por lo menos dos valores de tasa de descuento γ :
 - a) La política que siempre avanza.
 - b) La política aleatoria.

- 4. Implemente el algoritmo de iteración de valor y use su implementación para hallar la política óptima para por lo menos dos valores de tasa de descuento γ . Muestre la política óptima en cada caso (acción en cada casilla).
- 5. Repita el numeral anterior para la siguiente variación del juego:
 - Dos casillas azules elegidas *aleatoriamente* en las primeras dos filas del tablero.
 - Siete casillas rojas elegidas *aleatoriamente* en las primeras nueve filas del tablero.