**Requerimientos funcionales**

**Snake and Leaders**

**El programa esta en la capacidad de:**

* **Fundamentales**

**R#1 Jugar** el famoso juego de escalares y serpientes con las reglas estipuladas con la que cuenta este juego. Cada jugador debe de dar la indicación para que se juegue el siguiente turno a través de un salto de línea.

**R#2 Simular** el famoso juego de escalares y serpientes con las reglas estipuladas con la que cuenta este juego. Cada jugador debe de dar la indicación para que se juegue el siguiente turno a través de un salto de línea.

**R#3 Mostrar** una tabla de posiciones, en la cual, se muestren los mejores puntajes que obtienen los jugadores. Este puntaje se obtendrá de la siguiente manera, Numero de tiros por número de casillas (m\*n).

**R#4 Salir** del programa.

* **Gestión del juego**

**R#5 Crear** las estipulaciones necesarias para el correcto funcionamiento del juego, es decir, se ingresarán las reglas necesarias para crear el juego, el usuario ingresara una cadena con cinco elementos separados por un espacio, estos elementos serán y estarán organizados de la siguiente forma

**R#5.1 Crear** una matriz, la cual será el tablero de juego donde se realizarán las órdenes para que el juego funcione correctamente. Para crear la matriz utilizaremos los dos primeros elementos que se ingresaron en la cadena anterior, estos elementos serán y , donde el tamaño de la matriz está dada por la siguiente formula

**R#5.2 Crear** a los jugadores, si el elemento es un entero mayor que cero, se crearan los jugadores que indique este número con representación aleatoria en la cuadricula, de lo contrario si se ingresan símbolos, se crearan los jugadores según el rango de este símbolo, en otras palabras, si se ponen tres símbolos, se crearan 3 jugadores. Estos símbolos se deben de mantener en toda la partida.

**R#5.3 Crear** serpientes. Las serpientes en el juego unen una casilla con otra cualquiera en una fila inferior. Las serpientes se identifican con letras mayúsculas del alfabeto iniciando en A. Una serpiente puede ser representada a través de la letra que la identifica tanto en la casilla donde inicia como en la casilla donde termina. Las serpientes serán generadas de forma aleatoria. Restricciones: ninguna serpiente inicia en la casilla n x m, y ninguna casilla de inicio o fin de serpiente debe coincidir con otro inicio o fin de escalera o serpiente.

**R#5.4 Crear** escalera. Las escaleras en el juego unen una casilla con otra cualquiera en una fila superior. Las escaleras están numeradas desde 1 hasta la e, siendo cada número el identificador de dicha escalera. Una escalera puede ser representada a través del número que la identifica tanto en la casilla donde inicia como en la casilla donde termina. Las escalares serán generadas de forma aleatoria. Restricciones: ninguna escalera inicia en la casilla 1, y ninguna casilla de inicio o fin de escalera debe coincidir con otro inicio o fin de escalera o serpiente.

* **Gestión de la matriz**

**R#6 Manejar** la matriz de forma doblemente enlazada y de forma recursiva, donde no se usará ni un ciclo para crearla y cada elemento de esta matriz debe de conocer al elemento anterior, siguiente, arriba y abajo.

**R#7 Enumerar** la matriz, se empezará a enumerar la matriz donde el primer elemento será el número 1 y el ultimo será . Esta numeración empezará en la casilla o elemento inferior izquierda (Ultima fila y primera columna), esta tendrá el número 1, sigue en la casilla inmediatamente derecha y así hasta termina la fila, luego de esto sube en esa misma columna y se regresa a la izquierda, intercalando así la dirección en cada fila hasta llegar al elemento

**R#8 Mostrar** la matriz. Cuando se inicie el juego, el programa le mostrará una cuadrícula formada por corchetes, con las casillas numeradas correctamente y con la ubicación de las escaleras y las serpientes. Solo se mostrar la enumeración de las casillas una vez, cuando se inicia el juego.

* **Gestión de las indicaciones**

**R#9 Mover** a los jugadores creados, estos inician su recorrido a través del elemento enumerado como 1, moviéndose por turnos una cantidad de casillas igual a la mostrada por el dado lanzado en ese momento para ese jugador. Los jugadores se mueven en el mismo orden en que están numeradas las casillas. Al final gana el jugador que llegue primero a la casilla numerada con el número mayor.

* **Gestión de los jugadores**

**R#10 Representar** a los jugadores mediante la siguiente regla, si el ultimo elemento que se ingresó en la indicación del requerimiento 4

* **Durante el juego**

**R#11 Iniciar** el juego. Una vez se despliegue esta visualización de la cuadrícula, el programa queda esperando un salto de línea para iniciar y mostrar el primer tablero. Los tableros del juego irán mostrando la posición de los jugadores en las casillas, pero ya no deben mostrar los números de las casillas, aunque sí las escaleras y las serpientes. En adelante solo se podrá jugar cuando el usuario ingrese un salto de línea y jugará al que le corresponda el turno

**R#12 Lanzar** dado. Cada vez que sea el siguiente turno de un jugador, este dará un salto de línea lo cual indicara que se debe de lanzar un dado. Esto lo hará el programa internamente asignado un numero al azar entre 1 y 6, posteriormente dará un mensaje el cual indica el numero obtenido y después mostrar el tablero actualizado con la nueva posición del jugador.

**R#13 Subir** las escalares, cada vez que un jugador este sobre la casilla donde empieza una escalera, este se deberá mover a la casilla donde la escalera termina

**R#14 Bajar** las serpientes, cada vez que un jugador este sobre la casilla donde empieza una serpiente, este se deberá mover a la casilla donde termina la serpiente.

**R#15 Leer** palabra, cuando sea el turno de un jugador y este en vez de dar el salto de línea ingresa la palabra **num** y posteriormente un salto de línea se deberá de hacer el requerimiento R#8, es decir mostrar las cuadriculas numeradas, y la posición de las serpientes y escaleras.

* **Leer entradas**

**R#16 Leer** palabra, cuando sea el turno de un jugador y este en vez de dar el salto de línea ingresa la palabra **simul** y posteriormente un salto de línea el programa empezará en modo simulación, que consiste en ir mostrando lo que cada jugador a su turno juega, con el tablero correspondiente de cada nueva posición, esperando 2 segundos entre cada jugada, pero sin esperar ningún salgo de línea.

**R#13 Leer** palabra, cuando sea el turno de un jugador y este en vez de dar el salto de línea ingresa la palabra **menu** y posteriormente un salto de línea el juego se corta sin terminar y el programa regresa al menú principal, mostrándose sus opciones.

* **Terminar juego**

**R#14 Terminar** el juego. cuando uno de los jugadores llegue a la última casilla (la casilla cuya numeración es n **x** m). En ese caso debe decir: El jugador **Z** ha ganado el juego, con **Y** movimientos. Donde **Z** es el símbolo del jugador ganador y **Y** es la cantidad de veces que el jugador lanzó el dado en ese juego. A continuación de este mensaje, se pide el nombre o nickname del jugador ganador y posteriormente se muestra el menú principal del programa. Si se regresa al menú principal cuando algún jugador gana el juego, se calcula un puntaje para el usuario que es igual a la cantidad de movimientos multiplicado por la cantidad total de casillas del tablero. Este puntaje debe ser almacenado.

**R#15 Mostrar** puntaje, cuando se elija la opción mostrar puntaje se mostrará un listado de los nombres o nicknames de los jugadores, sus símbolos y sus respectivos puntajes, resultado de recorrer el árbol binario de búsqueda en inorden.