



Sistema Cine Bienestar U:
Especificación de requerimientos funcionales

Juan David Bahamon

Samuel Hernández

David Peñaranda

Algoritmos y programación II

Iván Darío Chacón Uribe

2022

Cliente	RICK
Usuario	Jugadores
Requerimientos funcionales	<ul style="list-style-type: none"> • Crear_Tablero
Requerimientos no funcionales	
Contexto del requerimiento	El programa debe pedir por consola N columnas, M filas y Q semillas para crear el tablero. Una vez ingresados los datos, se crea la estructura enlazada y pondrá al azar las semillas.

Cliente	RICK
Usuario	Jugadores
Requerimientos funcionales	<ul style="list-style-type: none"> • Enlaces
Requerimientos no funcionales	
Contexto del requerimiento	El programa debe pedir el número de enlaces P, este debe de ser menor a $0,5 \cdot (N \cdot M)$. Posteriormente se crean enlaces en la estructura enlazada de forma aleatoria, y en estas se debe evitar que una casilla se enlace con 2 o más casillas.

Cliente	RICK
Usuario	Jugadores
Requerimientos funcionales	<ul style="list-style-type: none"> • Ubicación_RickMorty
Requerimientos no funcionales	
Contexto del requerimiento	El programa debe de ubicar a Rick y Morty en alguna casilla de forma al azar.

Cliente	RICK
Usuario	Jugadores
Requerimientos funcionales	<ul style="list-style-type: none"> • Usernames
Requerimientos no funcionales	
Contexto del requerimiento	El programa debe de pedir los nombres de usuarios a los jugadores, quienes jugarán: el primero será de Rick y el segundo de Morty.

Cliente	RICK
Usuario	Jugadores
Requerimientos funcionales	<ul style="list-style-type: none"> • Menu_Opciones
Requerimientos no funcionales	
Contexto del requerimiento	Primero juega Rick, por lo que el programa debe desplegar un menú en el que el jugador podrá elegir: tirar el dado, ver tablero, ver enlaces o marcador.

Cliente	RICK
Usuario	Jugadores
Requerimientos funcionales	<ul style="list-style-type: none"> • Ver_Tablero
Requerimientos no funcionales	
Contexto del requerimiento	El programa debe mostrar la representación del tablero en ASCII, en caso de que el jugador lo desee.

Cliente	RICK
Usuario	Jugadores
Requerimientos funcionales	<ul style="list-style-type: none"> • Ver_Enlaces
Requerimientos no funcionales	
Contexto del requerimiento	El programa debe mostrar los enlaces en representación ASCII, usando la misma letra para las casillas que están enlazadas.

Cliente	RICK
Usuario	Jugadores
Requerimientos funcionales	<ul style="list-style-type: none"> • Ver_Marcador
Requerimientos no funcionales	
Contexto del requerimiento	El programa debe mostrar el marcador, que consiste en las semillas recolectadas por cada jugador.

Cliente	RICK
Usuario	Jugadores
Requerimientos funcionales	<ul style="list-style-type: none"> • Tirar_Dado
Requerimientos no funcionales	
Contexto del requerimiento	El jugador tira el dado, y el programa debe preguntar qué acción desea realizar, si avanzar o retroceder. En caso de caer en una super semilla se debe recolectar y actualizar el marcador. En caso de caer en una casilla con

	un portal, se debe trasportar a la casilla correspondiente, y si está contiene una super semilla, se recolecta.
--	-----------------------------------------------------------------------------------------------------------------

Cliente	RICK
Usuario	Jugadores
Requerimientos funcionales	<ul style="list-style-type: none"> Calcular_Ganador
Requerimientos no funcionales	
Contexto del requerimiento	El programa debe calcular el ganador con la siguiente fórmula: ganador=semillas recolectadas * 120 – tiempo en seg.

Cliente	RICK
Usuario	Jugadores
Requerimientos funcionales	<ul style="list-style-type: none"> Ver_Top5
Requerimientos no funcionales	
Contexto del requerimiento	El programa debe mostrar el top 5 de los mejores puntajes. Solo es permitido agregarse en el tablero a aquellos ganadores. En caso de repetir ganador y este está en el top 5, se acumularán los puntos que ha obtenido.

Cliente	RICK
Usuario	Jugadores
Requerimientos funcionales	
Requerimientos no funcionales	<ul style="list-style-type: none"> El tablero debe ser una estructura enlazada. Usar recursividad. Para los resultados usar arreglos, colecciones o lista. Almacenar puntajes.
Contexto del requerimiento	<p>El tablero debe ser una estructura enlazada. Se debe utilizar recursividad para recorrer el tablero.</p> <p>Para los resultados usar arreglos, colecciones o listas para implementar métodos de ordenamiento y búsqueda.</p> <p>Los puntajes se deben almacenar para no perder el histórico de puntajes.</p>

Nombre o identificador	Crear_Tablero		
Resumen	El programa debe pedir por consola N columnas, M filas y Q semillas para crear el tablero. Una vez ingresados los datos, se crea la estructura enlazada y pondrá al azar las semillas.		
Entradas	Nombre de entrada	Tipo de dato	Condición de selección o repetición
	N_Columnas	int	Ninguna
	M_Filas	int	El número de columnas ha sido ingresado correctamente.
	Q_Semillas	int	El número de filas ha sido ingresado correctamente.
Proceso	1) Ingresar número de columnas. 2) Ingresar número de filas. 3) Ingresar número de semillas. 4) Crear estructura enlazada. 5) Poner al azar las semillas.		
Resultado o postcondición	Lanzar una alerta en caso de ingresar mal los datos. En caso de que los datos sean permitidos, se creará el tablero.		
Salidas	Nombre de entrada	Tipo de dato	Condición de selección o repetición
	Mensaje de resultado	Alerta	Ninguna

Nombre o identificador	Enlaces
Resumen	El programa debe pedir el número de enlaces P, este debe de ser menor a $0,5*(N*M)$. Posteriormente se crean enlaces en la estructura enlazada de forma aleatoria, y en estas se debe evitar que una casilla se enlace con 2 o mas casillas.

Entradas	Nombre de entrada	Tipo de dato	Condición de selección o repetición
	P_Enlaces	double	El número es menor a $0,5*(N*M)$.
Proceso	1) Ingresar Número enlaces. 2) Validar que P sea un número menor a $0,5*(N*M)$. 3) Evitar que una casilla se enlace con 2 o más casillas. 4) Realizar enlaces aleatorios. 5) Crear enlaces en la estructura enlazada.		
Resultado o postcondición	En caso de que el número de enlaces sea invalido, evitar realizar enlaces y mostrar alerta. En caso de que los datos sean permitidos, se crean los enlaces.		
Salidas	Nombre de entrada	Tipo de dato	Condición de selección o repetición
	Mensaje de resultado	Alert	Ninguna

Nombre o identificador	Ubicación_RickMorty		
Resumen	El programa debe de ubicar a Rick y Morty en alguna casilla de forma al azar.		
Entradas	Nombre de entrada	Tipo de dato	Condición de selección o repetición
	Nombre	String	Ninguna
	Ninguna	Ninguna	Ninguna
Proceso	1) Ubicar a Rick en alguna casilla. 2) Ubicar a Morty en alguna casilla.		
Resultado o postcondición	Se ha ubicado correctamente a Rick y Morty en alguna de las casillas.		

Salidas	Nombre de salida	Tipo de dato	Condición de selección o repetición
	Mensaje de resultado	Alerta	Ninguna

Nombre o identificador	Username		
Resumen	El programa debe de pedir los nombres de usuarios a los jugadores, quienes jugarán: el primero será de Rick y el segundo de Morty.		
Entradas	Nombre de entrada	Tipo de dato	Condición de selección o repetición
	Username_Rick	String	Ninguna
	Username_Morty	String	Se ha ingresado el nombre de Rick.
Proceso	1) Ingresar nombre de usuario de quien jugará con Rick. 2) Ingresar nombre de usuario de quien jugará con Morty.		
Resultado o postcondición	Se asignan los usernames a cada personaje.		
Salidas	Nombre de entrada	Tipo de dato	Condición de selección o repetición
	Ninguna	Ninguna	Ninguna

Nombre o identificador	Menu_Opciones
------------------------	----------------------

Resumen	Primero juega Rick, por lo que el programa debe desplegar un menú en el que el jugador podrá elegir: tirar el dado, ver tablero, ver enlaces o marcador.		
Entradas	Nombre de entrada	Tipo de dato	Condición de selección o repetición
	opcion	int	Ninguna
Proceso	1) El jugador ingresa una opción.		
Resultado o postcondición	El programa procede a realizar la opción correspondiente.		
Salidas	Nombre de entrada	Tipo de dato	Condición de selección o repetición
	Mostrar	String/accion	Se ha seleccionado una opción

Nombre o identificador	Ver_Tablero		
Resumen	El programa debe mostrar la representación del tablero en ASCII, en caso de que el jugador lo desee.		
Entradas	Nombre de entrada	Tipo de dato	Condición de selección o repetición
	opcion	int	Se ha ingresado el número 2
Proceso	1) Ingresar número 2. 2) Mostrar representación del tablero.		
Resultado o postcondición	Mostrar representación en ASCII del tablero.		
Salidas	Nombre de entrada	Tipo de dato	Condición de selección o repetición

	mostrarTablero	String	Ninguna
--	----------------	--------	---------

Nombre o identificador	Ver_Enlaces		
Resumen	El programa debe mostrar los enlaces en representación ASCII, usando la misma letra para las casillas que están enlazadas.		
Entradas	Nombre de entrada	Tipo de dato	Condición de selección o repetición
	opcion	int	Se ha ingresado el número 3
Proceso	1) Ingresar número 3. 2) Mostrar enlaces del tablero.		
Resultado o postcondición	Mostrar enlaces en ASCII del tablero.		
Salidas	Nombre de entrada	Tipo de dato	Condición de selección o repetición
	mostrarEnlaces	String	Ninguna

Nombre o identificador	Ver_Marcador		
Resumen	El programa debe mostrar el marcador, que consiste en las semillas recolectadas por cada jugador.		

Entradas	Nombre de entrada	Tipo de dato	Condición de selección o repetición
	opcion	int	Se ha ingresado el número 4
Proceso	1) Ingresar número 4. 2) Mostrar marcador de cada jugador.		
Resultado o postcondición	Mostrar marcador de cada jugador.		
Salidas	Nombre de entrada	Tipo de dato	Condición de selección o repetición
	mostrarMarcador	String	Ninguna

Nombre o identificador	Tirar_Dado		
Resumen	El jugador tira el dado, y el programa debe preguntar qué acción desea realizar, si avanzar o retroceder. En caso de caer en una super semilla se debe recolectar y actualizar el marcador. En caso de caer en una casilla con un portal, se debe transportar a la casilla correspondiente, y si está contiene una super semilla, se recolecta.		
Entradas	Nombre de entrada	Tipo de dato	Condición de selección o repetición
	opcion	int	Se ha ingresado el número 1
	movimiento	int	Se ha ingresado un número correcto.
Proceso	1) Ingresar número 1. 2) El programa pregunta si desea avanzar o retroceder. 3) Mover el personaje a la casilla correspondiente por la tirada del dado. 4) En caso de caer en una super semilla se debe recolectar.		

	5) En caso de caer en un portal, se debe trasladar a la casilla correspondiente. 6) En caso de recolectar una super semilla se debe actualizar el marcador.		
Resultado o postcondición	Se ha realizado el movimiento de los personajes. En caso de caer en un portal, en una semilla o ambos, se atrapa la semilla o se traslada a otra casilla, y por último se actualiza el marcador		
Salidas	Nombre de entrada	Tipo de dato	Condición de selección o repetición
	Mensaje	Alert	Recolecta una semilla o cae en un portal.

Nombre o identificador	Calcular_Ganador		
Resumen	El programa debe calcular el ganador con la siguiente fórmula: ganador=semillas recolectadas * 120 – tiempo en seg.		
Entradas	Nombre de entrada	Tipo de dato	Condición de selección o repetición
	Ninguna	Ninguna	Ninguna
Proceso	1) Se obtienen las semillas recolectadas por cada jugador. 2) Se calcula el puntaje para obtener el ganador.		
Resultado o postcondición	Se obtiene el ganador.		
Salidas	Nombre de entrada	Tipo de dato	Condición de selección o repetición
	mostrarGanador	String	Ninguna

Nombre o identificador	Ver_Top5		
Resumen	El programa debe mostrar el top 5 de los mejores puntajes. Solo es permitido agregarse en el tablero a aquellos ganadores. En caso de repetir ganador y este está en el top 5, se acumularán los puntos que ha obtenido.		
Entradas	Nombre de entrada	Tipo de dato	Condición de selección o repetición
	nombreGanador	String	Es ganador.
	puntajeGanador	int	Es ganador.
Proceso	1) Obtener nombre y puntaje del ganador. 2) Ingresar en el top 5 de los mejores jugadores. 3) En caso de estar en el top, acumular puntos obtenidos. 4) Actualizar tabla del top 5.		
Resultado o postcondición	Mostrar top 5 de los mejores jugadores.		
Salidas	Nombre de entrada	Tipo de dato	Condición de selección o repetición
	mostrarTop5	String	Ninguna

