



Algoritmos y Estructuras de Datos I

Área de Ingeniería en Computadores

Semestre II

Grupo 1

2021

Proyecto 2: *Let's play!*

Estudiantes

Carné

Juan Peña Rostrán

2018080231

Josué Guillermo Granados Chacón

2020072009

Anthony Montero Román

2019275097

Profesor:

Antonio González Torres

Índice

Introducción.....	4
Diseño del proyecto	5
Metas del Proyecto:	5
Listado de Requerimientos:	5
Requerimientos para BP Game:	5
Genetic Puzzle:	6
Roles:	6
Reglas:	6
Cronograma:	7
Minutas de las sesiones de trabajo:	8
Bitácora:	8
Referencias	13

Introducción

El presente proyecto muestra un videojuego llamado *Let's play*, el cual hace uso de estrategias para el *pathfinding* y procesamiento de imágenes, como lo son los algoritmos de backtracking y algoritmos genéticos. Por un lado, los algoritmos de *backtracking* (vuelta atrás) se presentan como una herramienta para solucionar problemas en particular. En general, se hace un estudio de todas las posibles soluciones para el problema, o si en definitiva no hay solución, esto lleva a que sea de naturaleza recursiva, esto se logra con la condición de parada, es decir, ¿qué se hace cuando se acaba la búsqueda? Si hubo solución, retornar dicha solución, sino se hace el retroceso de cada subtarea a la original para buscar otro camino (Valbuena, Torres y Zapata, 2008).

En este caso, se pretende emular un tablero de “fútbolín” donde un número determinado de jugadores estáticos tienen un tiro para anotar en la meta del otro equipo. El jugador puede ver la fuerza y dirección del tiro con una guía, además de poder ver el camino más corto hacia la meta desde su ubicación.

Por otro lado, los algoritmos genéticos se encuentran fundamentados en la evolución de las poblaciones mediante la selección natural y cómo progresan hacia mejores regiones donde el espacio de búsqueda es óptimo. Para que una población del algoritmo evolucione se aplican operadores genéticos bajo los conceptos probabilísticos presentes en selección, recombinación y mutación de dicha población (Valencia, 1997). Para este proyecto, se requirió de la implementación de un algoritmo genético para el procesamiento de imágenes en un rompecabezas, donde cada avance en la imagen es más cercano a la original, el cual está controlado por botones estilo reproductor musical.

Diseño del proyecto

Metas del Proyecto:

Se debe generar un documento para la demostración de evidencias durante el desarrollo del proyecto.

Se debe generar los repositorios correspondientes en GitHub para los programas *Server* y *Client*.

Se debe generar la documentación externa del proyecto en la sección Wiki de GitHub y la interna en Doxygen.

Se debe implementar el algoritmo de *pathfinding* A* para indicar a los jugadores el camino más corto a la meta.

Se debe implementar un algoritmo de backtracking para encontrar el camino más corto a la meta cuando se juega contra la máquina.

Se debe implementar un algoritmo genético para la resolución de un rompecabezas asociado a una imagen específica.

Listado de Requerimientos:

Requerimientos para BP Game:

1. Yo como usuario quiero elegir el número máximo de jugadores por equipo para empezar una partida.
2. Yo como usuario quiero definir el número de goles necesarios para ganar una partida.
3. Yo como usuario deseo patear el balón con una fuerza determinada para hacer los goles
4. Yo como usuario deseo patear el balón con una dirección determinada para hacer los goles
5. Yo como usuario quiero ver la ruta de la bola una vez lanzada para analizar los siguientes tiros
6. Yo como usuario quiero tener la posibilidad de elegir un contrincante entre otro jugador o la máquina

7. Yo como usuario deseo conocer la ruta más corta hacia donde el balón podría ingresar al marco rival para anotar goles
8. Yo como usuario deseo ubicar aleatoriamente los jugadores del equipo en un tablero para que tengan una ubicación fija.
9. Yo como usuario deseo ver un marcador para conocer el estado de la partida.

Genetic Puzzle:

10. Yo como usuario quiero poder cargar una imagen al sistema para que sea procesada.
11. Yo como usuario quiero elegir en cuántos fragmentos se va a dividir la imagen para observar el rompecabezas.
12. Yo como usuario deseo que los fragmentos de la imagen se acomoden de forma aleatoria.
13. Yo como usuario quiero que el sistema acomode la imagen en su forma original de manera incremental para conocer cómo resuelve el rompecabezas.
14. Yo como usuario deseo tener disponible una barra que muestre el flujo en que avanza la solución del rompecabezas, con las opciones de retroceder, pausar y avanzar como un reproductor de música.

Roles:

Scrum Master: Juan Peña

Product Owner: Anthony Montero

Developers: Anthony Montero, Josué Granados, Juan Peña

Reglas:

1. Dar seguimiento al cronograma.
2. Comentar internamente el código cada vez que se actualiza una sección.
3. Las reuniones se realizan por Discord
4. Se designa a Juan Peña y Josué Granados como encargados del Backend
5. Se designa a Anthony Montero como encargado del Frontend.

Cronograma:

Actividades programadas	Responsables	Fecha estimada
Discusión de los requerimientos del proyecto, creación de los repositorios respectivos en Github.	Anthony Montero, Josué Granados, Juan Peña	13/5/2021
Definir el framework donde se desarrollará la interfaz de usuario.	Anthony Montero, Josué Granados, Juan Peña	14/5/2021
Creación del server, comunicación con sockets y serialización en formato JSON para manejo de objetos.	Josué Granados, Juan Peña	16/5/2021
Tener algoritmo pathfinding A* y Backtracking implementado	Josué Granados, Juan Peña	19/5/2021
Diseñar el esqueleto a utilizar para la Interfaz de Usuario (UI)	Anthony Montero	19/5/2021
Presentar segunda versión del esqueleto de la UI (tablero de juego dividido en cuadros y personajes estáticos)	Anthony Montero	23/5/2021
Habilitar el procesamiento de la posición de los jugadores, la bola y el tablero de juego en el servidor	Josué Granados, Juan Peña	23/5/2021

Minutas de las sesiones de trabajo:

Bitácora:

Fecha: 13-5-2021

Horas: 5:30 – 8:00 pm

Actividades: Se descompuso los requerimientos del proyecto en *user stories* y *features*. Además, se definieron los roles de los participantes. *Project Owner*: Anthony Montero, *Scrum Master*: Juan Peña. *Developers*: Anthony Montero, Josué Granados, Juan Peña. Se definió una primera versión del cronograma de trabajo, así como los responsables.

Responsables: Anthony Montero, Josué Granados, Juan Peña

Fecha: 14-5-2021

Horas: 8:30 – 10:00 pm

Actividades: Se descompuso las historias de usuario en tareas para realizar el plan de iteraciones en la sección “*Projects*” de *Github*.

Responsables: Juan Peña

Fecha: 16-5-2021

Horas: 8:30 – 10:00 pm

Actividades: Se agregó una primera versión para hacer búsqueda de caminos con backtracking en una matriz de 4x4 desde la posición (0, 0) hasta la (3, 3), donde los movimientos permitidos fueron: izquierda, abajo y derecha

Responsables: Juan Peña

Fecha: 17-5-2021

Horas: 8:00 – 10:00pm

Actividades: Se crea un servidor TCP en C++.

Responsables: Anthony Montero

Fecha: 18-5-2021

Horas: 8:30 – 12:00 pm

Actividades: Se agregó una segunda versión más robusta para buscar un camino en una matriz 10x10 desde un punto (x1, y1) hasta un punto (x2, y2), sin embargo, solo se pudo obtener la cantidad de bloques que componen el camino y no las posiciones que es lo que se necesita.

Responsables: Juan Peña

Fecha: 20-5-2021

Horas: 10:00 – 12:00pm

Actividades: Se crea el cliente en el programa QTCreator.

Responsables: Anthony Montero

Fecha: 24-5-2021

Horas: 8:00 – 12:00pm

Actividades: Se investiga acerca de bibliotecas para el procesamiento de imágenes.

Responsables: Juan Peña

Fecha: 25-5-2021

Horas: 7:00 – 1:00am

Actividades: Se intenta instalar la biblioteca openCV el cual resulta fallido.

Responsables: Juan Peña

Fecha: 27-5-2021

Horas: 8:00 – 12:00pm

Actividades: Se intenta otro método de instalación de la biblioteca el cual se interrumpe debido a que se fue la conexión a internet

Responsables: Juan Peña

Fecha: 29-5-2021

Horas: 6:00 – 10:00pm

Actividades: Se crea el esqueleto de la interfaz del juego en QTCreator, con imágenes y demás.

Responsables: Anthony Montero

Fecha: 30-5-2021

Horas: 5:00 – 11:00pm

Actividades: Se crean las diferentes clases para los jugadores, el balón, y los marcos. Se crean la lógica para las posiciones y el movimiento del balón.

Responsables: Anthony Montero

Fecha: 30-5-2021

Horas: 8:00 pm– 1:00am

Actividades: Se crea la clase JsonParser que se encarga de serializar y deserializar mensajes en formato JSON para mantener el flujo de eventos en el juego.

Responsables: Juan Peña

Fecha: 30-5-2021

Horas: 8:00 pm– 1:00am

Actividades: Se crea la clase JsonParser que se encarga de serializar y deserializar mensajes en formato JSON para mantener el flujo de eventos en el juego.

Responsables: Juan Peña

Fecha: 1-6-2021

Horas: 2:00 pm– 4:00am

Actividades: se trabaja en la documentación externa, precisamente la introducción de este documento y la organización de repositorio.

Responsables: Juan Peña

Fecha: 3-6-2021

Horas: 10:00 pm– 1:00am

Actividades: se trabaja en la corrección de errores en el manejo de los archivos JSON

Responsables: Juan Peña

Fecha: 4-6-2021

Horas: 10:00 pm– 1:00am

Actividades: se cambia el formato de la lista enlazada y se especializa de genérica a que sea exclusivamente para el objeto jugador. Se agrega el método en el servidor que procesa los mensajes para definir cuál algoritmo se debe ejecutar.

Responsables: Juan Peña

Fecha: 5-6-2021

Horas: 9:00 pm– 2:00am

Actividades: se trabaja en la corrección de errores en el manejo de los archivos JSON y se integra el JsonParser con el server para la lectura de la configuración del juego

Responsables: Juan Peña

Fecha: 5-6-2021

Horas: 2:00 – 11:00pm

Actividades: Se crea la lógica para el envío de datos, clase para formar el json y las colisiones.

Responsables: Anthony Montero

Fecha: 6-6-2021

Horas: 11:00 am– 8:00pm

Actividades: se trabaja en las formas de envío de información al server, específicamente en cómo se envía el camino más corto encontrado cuando se ejecuta el algoritmo de pathfinding. Se actualiza el repositorio, específicamente el plan de iteraciones, la wiki y el readme.

Responsables: Juan Peña

Fecha: 12-6-2021

Horas: 1:00 – 11:00pm

Actividades: Se crea una clase para formar botones, la función de estos, se crea la lógica para la fuerza de tiro, la clase para el marcador y la lógica para estos.

Responsables: Anthony Montero

Fecha: 13-6-2021

Horas: 5:00 – 10:00pm

Actividades: Se ajusta la lógica de algunas clases, se arregla las posiciones de los jugadores y se prueba la respuesta del servidor.

Responsables: Anthony Montero

Fecha: del 12 al 15 de junio

Actividades: se trabajó en la corrección de errores en el algoritmo de backtracking, además de reemplazar el algoritmo de pathfinding A*. Se reajustaron los valores en el tamaño de campo de juego y se definió que el archivo Json que guarda el camino más corto sea limpiado cada cambio de turno en el juego. Se integró los algoritmos con el servidor según sus condiciones. Se agregó documentación externa en la Wiki y por medio de Doxyfile. Finalmente se reajustó el diagrama de clases.

Responsables: Juan Peña

Referencias

Valbuena, S., Torres, S. & Zapata, D (2008). Introducción a las Estructuras de datos en Java. ELIZCOM SAS.

Valencia, P. E. (1997). Optimización mediante algoritmos genéticos. In Anales del Instituto de Ingenieros de Chile (Vol. 109, No. 2, pp. 83-92).