

## Proyecto II – PacManCE

Instituto Tecnológico de Costa Rica  
Área Académica Ingeniería en Computadores  
CE2103 - Algoritmos y Estructuras de Datos II  
Primer Semestre 2023  
Valor: 20%



### Objetivo General

- Implementar una solución que permita la resolución del contexto descrito en esta especificación utilizando el lenguaje de programación C++, desarrollo mobile, el algoritmo Pathfinding y la técnica de diseño de algoritmos Backtracking.

### Objetivos Específicos

- Analizar los requerimientos establecidos en esta especificación para el desarrollo de un juego utilizando el Paradigma Orientado a Objetos, el lenguaje de programación C++, desarrollo mobile, el algoritmo Pathfinding y la técnica de diseño de algoritmos Backtracking.
- Diseñar un juego que permita el cumplimiento de los requerimientos establecidos en esta especificación utilizando el Paradigma Orientado a Objetos, el lenguaje de programación C++, desarrollo mobile, el algoritmo Pathfinding y la técnica de diseño de algoritmos Backtracking.
- Implementar un juego que permita el cumplimiento de los requerimientos establecidos en esta especificación utilizando el Paradigma Orientado a Objetos, el lenguaje de programación C++, desarrollo mobile, el algoritmo Pathfinding y la técnica de diseño de algoritmos Backtracking.

### Descripción del Problema

Deben implementar una reproducción del juego Pac-Man, el cual consiste en un videojuego, tipo arcade, lanzado a inicios de los años 80. Esta implementación debe realizarse mediante una aplicación de escritorio y una aplicación mobile.



Como descripción general, este videojuego consiste en un personaje circular y amarillo, el cual debe recorrer un tablero y comer una serie de puntos, los cuales se encuentran distribuidos en el tablero. Para esto debe evadir una serie de enemigos, los cuales corresponden a cuatro fantasmas, que tienen como objetivo cazarlo, y así evitar que pueda comer todos los puntos del tablero. En el siguiente link pueden acceder a una versión del juego que pueden utilizar como referencia: <https://g.co/kgs/ijveVX>

Para el desarrollo del videojuego deben ser implementadas las siguientes funcionalidades, algunas de las cuales son adaptaciones para cumplir con fines académicos del curso:

1. El juego debe desarrollarse en un tablero, donde algunas celdas corresponden a espacios que permiten al personaje desplazarse, mientras que otras celdas corresponden a obstáculos, los cuales impiden su desplazamiento. En cada celda, que no represente un obstáculo, debe haber un punto que el personaje pueda comer.
2. El juego debe permitir, al menos, cuatro niveles, donde cada nivel debe tener una organización de celdas diferentes, es decir, obstáculos distribuidos de forma diferente en cada nivel.
3. Posibilidad de que el personaje pueda ir comiendo los puntos según el desplazamiento que vaya realizando. Por cada punto que el personaje coma, el jugador acumula 10pts en su puntaje total. El desplazamiento lo realiza el jugador utilizando el teclado de la computadora.
4. El personaje solo puede moverse de manera horizontal y vertical, una celda a la vez. No está permitido el movimiento en diagonal.
5. Existen enemigos, fantasmas, que persiguen al personaje. El objetivo de estos enemigos es cazar al personaje y así evitar que se coma los puntos del tablero.
6. En el primer nivel solo debe aparecer un enemigo. Conforme avanza de nivel se debe incorporar un nuevo enemigo, hasta alcanzar el máximo de cuatro enemigos. Si se implementaran más de cuatro niveles, después del cuarto nivel siempre habrían cuatro enemigos.
7. Cada vez que un enemigo alcance al personaje, se pierde una vida.
8. Cuando el personaje pierde una vida debe ser colocado en una posición aleatoria, lejos de los enemigos, para que pueda continuar con la misión de comer todos los puntos del tablero.
9. Para efectos del juego, el personaje tiene tres vidas. Al perder las tres vidas termina el juego.
10. Otra forma de terminar el juego es que se completen todos los niveles, para lo cual, al final, debe mostrar el puntaje total obtenido por el jugador.
11. Cada 200pts que acumule un jugador, se colocará en el tablero un poder que le permite al personaje invertir roles e intentar cazar a los enemigos.
12. Por cada enemigo cazado se le suma 50pts al puntaje total del jugador.
13. Cuando un enemigo es cazado, desaparece del tablero y 5 segundos después vuelve a aparecer en una posición aleatoria del tablero.
14. Cuando aparece el poder en el tablero, los enemigos están en la posibilidad de robar el poder al personaje.
15. Cuando aparece el poder en el tablero, la mitad de los enemigos deben usar Pathfinding A\* para encontrar la ruta más corta hacia el poder y robárselo al personaje. En consola debe mostrar el resultado de cada iteración del algoritmo, hasta encontrar la ruta seleccionada. Para esto último, por cada iteración, debe mostrar la celda seleccionada, las celdas que se encuentren en la Open-List y las celdas que se encuentren en la Closed-List. Esto debe mostrarlo para cada enemigo al cual se le asignó este algoritmo.
16. Cuando aparezca el poder en el tablero, la otra mitad de los enemigos deben usar Backtracking para encontrar la ruta más corta hacia el poder y robárselo al personaje. Para esto último, en consola debe mostrar el árbol de decisiones que utilizó el algoritmo para encontrar la ruta. Esto debe mostrarlo para cada enemigo al cual se le asignó este algoritmo.
17. En la pantalla del juego se debe mostrar:
  - a. Puntaje total del jugador en tiempo real, durante todo el desarrollo de la partida.
  - b. Cantidad de vidas restantes del jugador.
  - c. Nivel en el que se encuentra el jugador.

## Controller

Se debe implementar una aplicación mobile que permita lo siguiente:

- Utilizar el giroscopio del celular para controlar el movimiento del personaje en el tablero.
- Mostrar el puntaje total del jugador en tiempo real, durante todo el desarrollo de la partida.
- Mostrar la cantidad de vidas restantes del jugador.
- Mostrar el nivel en el que se encuentra el jugador.

*NOTA: Estas funcionalidades deben ser implementadas a través de la aplicación mobile, adicional a que estas mismas funcionalidades deben ser implementadas también en la aplicación de escritorio.*

Finalmente, es importante recordar que, cualquier algoritmo o estructura de datos que se requiera para la implementación del proyecto, debe ser implementada por los integrantes del grupo. Es decir, no está permitido utilizar bibliotecas para estos fines.

## Documentación requerida

1. Internamente, el código se debe documentar siguiendo los estándares de documentación para cada lenguaje utilizado.
2. La documentación externa deberá desarrollarse en Latex utilizando la plantilla llamada IEEE Conference Template, que se puede descargar desde este link: <https://www.overleaf.com/gallery/tagged/ieee-official/page/1>. Para el desarrollo en Latex se recomienda el uso de Overleaf. La documentación debe contener las siguientes secciones:
  - a. Breve descripción del problema.
  - b. Diagrama de clases.
  - c. Explicación del algoritmo A\* y Backtracking implementado
  - d. Problemas encontrados: En esta sección se detalla cualquier problema que no se ha podido solucionar en el trabajo.
3. **Planificación y administración del proyecto:** se utilizará Jira para la administración del proyecto. Debe incluir:
  - a. Lista de features e historias de usuario identificados de la especificación.
  - b. Plan de iteraciones que agrupen cada bloque de historias de usuario de forma que se vea un desarrollo incremental
  - c. Descomposición de cada user story en tareas.
  - d. Asignación de tareas entre los integrantes del grupo

## Aspectos operativos y evaluación:

1. **Fecha de entrega: De acuerdo con el cronograma del curso y lo establecido en el TEC Digital**
2. El proyecto tiene un valor de 20% de la nota del curso.
3. El trabajo es **en parejas**.
4. Deben entregar un documento con el link del repositorio de GitHub, Jira y el PDF de la documentación. En ambas herramientas deben dar acceso al correo del profesor.
5. Es obligatorio utilizar un Git y GitHub para el control de versiones del código fuente y evidenciar el uso de Commits frecuentes.
6. Es obligatorio integrar toda la solución.
7. La aplicación desktop tendrá un valor total de 60%, la implementación de la aplicación mobile un valor de 20%. La documentación externa y el anexo tendrá un valor de 20%.
8. De las notas mencionadas en el punto anterior se calculará la Nota Final del Proyecto.
9. Se evaluará que la documentación sea coherente, acorde a la dificultad/tamaño del proyecto y el trabajo realizado. Se recomienda que realicen la documentación conforme se implementa el código.

10. La nota de la documentación externa es proporcional a la completitud del proyecto.
11. Las citas de revisión oficiales serán determinadas por el profesor durante las lecciones o mediante algún medio electrónico.
12. Los estudiantes pueden seguir trabajando en el código hasta 15 minutos antes de la primera cita de revisión oficial.
13. Aún cuando el código y la documentación externa tienen sus notas por separado, se aplican las siguientes restricciones
  - a. Si no se entrega documentación externa, automáticamente se obtiene una nota de cero en la nota final del proyecto.
  - b. Si no se utiliza un manejador de código se obtiene una nota de cero en la nota final del proyecto.
  - c. Si la documentación externa no se entrega en la fecha indicada se obtiene una nota de cero en la nota final del proyecto.
  - d. Si el código no compila se obtendrá una nota de cero en la nota final del proyecto, por lo cual se recomienda realizar la defensa con un código funcional.
  - e. El código debe ser desarrollado en C++ (Linux), en caso contrario se obtendrá una nota de cero en la nota final del proyecto.
  - f. La aplicación mobile debe ser desarrollada nativamente o no mediante alguna herramienta que permita crear sitios webs que se adapten o desplieguen en un dispositivo mobile, caso contrario se obtendrá una nota de cero en la nota final del proyecto.
14. La revisión de la documentación podría ser revisada antes, durante o después de la cita de revisión del proyecto.
15. Cada grupo tendrá como máximo 15 minutos para exponer su trabajo al profesor y realizar la defensa de éste. Es responsabilidad de los estudiantes mostrar todo el trabajo realizado, por lo cual se recomienda tener todo listo antes de ingresar a la defensa.
16. Cada excepción o error que salga durante la ejecución del proyecto y que se considere debió haber sido contemplada durante el desarrollo del proyecto, se castigará con 2 puntos de la nota final del proyecto.
17. Cada grupo es responsable de llevar los equipos requeridos para la revisión, si no cuentan con estos deberán avisar al menos 2 días antes de la revisión al profesor para coordinar el préstamo de estos.
18. Durante la revisión únicamente podrán participar el estudiante, asistentes, otros profesores y el coordinador del área.

## **ANEXO DEL PROYECTO**

### **Objetivo General**

- Aplica el conocimiento y la comprensión de los principios de gestión de ingeniería y la toma de decisiones económicas y los aplica al propio trabajo, como miembro y líder de un equipo, y para gestionar proyectos en entornos multidisciplinarios.

### **Objetivos Específicos**

- Identificar el ciclo de vida de un proyecto, así como otros elementos relacionados con la gestión del proyecto de ingeniería.
- Identificar los diversos tipos de beneficios, costos económicos y financieros de una actividad de ingeniería.
- Planificar el proceso y los elementos financieros asociados a la gestión del proyecto de ingeniería, considerando entornos multidisciplinarios.
- Aplicar herramientas y elementos de la gestión de proyectos y finanzas para el logro de las metas establecidas.

### **Atributos de Acreditación**

- Administración de Proyectos y Finanzas.

### **Descripción del Entregable**

Cada grupo debe elaborar un documento que tenga la siguiente estructura:

1. Portada.
2. Tabla de contenidos.
3. Introducción.
4. Ciclo de vida del proyecto. Describir la relación que existe entre un modelo de desarrollo ágil y la estrategia solicitada para el desarrollo de este proyecto.
5. Beneficios, costos económicos y financieros. Describir las ventajas que tuvo la estrategia solicitada para el desarrollo de este proyecto, una vez finalizado el mismo. También deben indicar el costo que implicaría lanzar este juego al mercado. Para esto último se recomienda considerar el costo del hosting de la aplicación, así como el costo de lanzarlos para las plataformas Android y iOS.
6. Proceso y elementos financieros de la gestión del proyecto. Describir el plan de iteraciones elaborado para el proyecto, así como un listado de reglas para la gestión del trabajo en equipo.
7. Herramientas y elementos de la gestión de proyectos y finanzas. Indicar las herramientas utilizadas para el desarrollo del proyecto

### **Aspectos operativos y evaluación**

1. El documento debe ser enviado en formato PDF en conjunto con los entregables del proyecto.