

UPC: CC184 - Complejidad Algorítmica

Trabajo Parcial: Quoridor

Profesores:

Canaval Sanchez, Luis Martín
Sopla Maslucan Abraham
Ricardo Eugenio González Valenzuela

14/09/2020

1. Objetivo

El presente documento define el trabajo parcial y la rúbrica que permite evaluar el logro del curso CC76 Complejidad Algorítmica. El objetivo del trabajo parcial (TP) es que los estudiantes apliquen los conocimientos adquiridos de complejidad algorítmica, así como de enfoques y paradigmas de programación, tomando conciencia de la importancia de los algoritmos en la industria percibiéndolos como tecnología en lugar de como simples rutinas de procesos.

1.1. Logro del curso

Diseña algoritmos eficientes tomando en cuenta el tiempo, el espacio y la complejidad.

- En Ingeniería de Software, el logro contribuye a alcanzar el: **ABET – EAC - Student Outcome 4**: La capacidad de reconocer responsabilidades éticas y profesionales en situaciones de ingeniería y hacer juicios informados, que deben considerar el impacto de las soluciones de ingeniería en contextos globales, económicos, ambientales y sociales.
- En Ciencias de la Computación, el logro contribuye a alcanzar el: **ABET – CAC - Student Outcome 4**: Reconocer las responsabilidades profesionales y tomar decisiones informadas sobre prácticas de computación basadas en principios legales y éticos.

2. Enunciado

2.1. Reseña

Quoridor es un juego donde el objetivo de cada participante es llegar hasta la base del rival, a cada turno se debe escoger entre colocar dificultades como muros o avanzar, lo mas probable es que pierdas si te mantienes enfocado en una acción específica.

Para esto, el juego cuenta con:

- Un tablero de 9×9 recuadros;
- 4 fichas (una por jugador);
- 20 paredes (tablas que tapan dos casillas, se repartiran equitativamente entre el número de jugadores)

En el siguiente link puede encontrar un [breve video-tutorial de Quoridor](#)

2.2. Mecánica

El objetivo del juego es ser el primero en llegar a cualquier casilla del extremo opuesto del tablero, *i.e.*: el extremo opuesto a donde inicia el jugador. Si se desea ser mas estricto, podría restringir la llegada únicamente a la casilla central del extremo opuesto.

En su turno cada jugador deberá tomar una decisión:

1. mover un espacio de forma vertical u horizontal (salvo excepciones), o
2. colocar una barrera (cada barrera bloquea dos cuadros y debe colocarse de manera que lo haga perfectamente para evitar confusiones).

En caso que un jugador quede frente a otro, éste en su turno puede escoger saltar por encima del primero, en caso de no ser posible, entonces podría mover un espacio en diagonal. Note que de esta forma puede estar ejecutando un movimiento ventajoso (por ser de dos casillas).

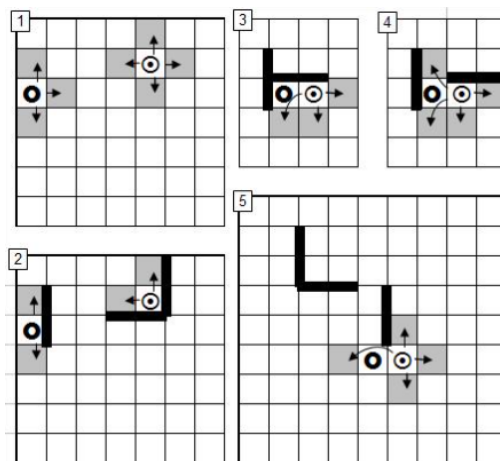


Figura 1: Movimientos posibles considerando posiciones de muros y rivales (Imagen obtenida de [cuevadelobo.com](#))

Observaciones: Ningún recuadro debe quedar inaccesible por causa de las barreras. De lo anterior, deducimos que ningún jugador podrá quedar encerrado entre barreras,

3. Cronograma

. **Semana 5:** Selección de grupos (conformados por 3 integrantes, como máximo puede existir un grupo de 2 integrante). Envío de las secciones de: *Introducción*, *Estado del arte* y *Metodología* del documento final. Entrega: hasta el final de la semana 5, sábado 23:59:59 horas.

. **Semana 6:** En la primera clase:

- a) declaración de 3 algoritmos a usar para alcanzar el objetivo;
- b) así mismo, identificación del espacio de búsqueda;
- c) declaración de diferentes tamaños de entradas de datos para trabajar;
- d) declaración de las métricas que se usarán para evaluar la eficiencia de los algoritmos a usar.

Al finalizar la semana, sábado 23:59:59 horas, cada grupo debe enviar los algoritmos implementados adecuadamente, y alguna prueba mínima del funcionamiento de los mismo. Aún no será necesario enviar las métricas y demás.

. **Semana 7:** Presentación, de máximo 7 minutos por grupo, de la investigación realizada, finalizar todos los puntos declarados en la semana 6 demostrando haber cumplido con los objetivos del trabajo. Entrega: hasta una hora antes de la última clase de la semana 7. Presentación: durante la última clase de la semana 7.

Observaciones:

- Para el trabajo parcial, no será necesario implementar la funcionalidad de las barreras.
- En caso de existir mas de un grupo de 2 alumnos, el profesor tiene libertad de reasignar alumnos arbitrariamente.
- La presentación de la semana 7, no deberán exceder por ninguna razón el límite de tiempo indicado. Los integrantes deben preocuparse por controlar este tiempo, de otra forma la exposición será interrumpida y repercutirá negativamente sobre la nota del grupo.

4. Entrega

- El trabajo será colocado en una cuenta de [Github](#)
- Ningún *commit* realizado luego de la hora de entrega será revisado.
- Se revisará que todos los integrantes tengan participación en los *commit*, y cual fue el grado en que participaron.
- La descripción de cada *commit* debe ser clara y concisa.
- Al aula virtual sólo se ha de enviar el link al repositorio de [Github](#) que contenga el trabajo.

- El repositorio ha de contener obligatoriamente:
 - un archivo [IPython Notebook](#). Explicando detalladamente cada una de los conceptos indicados en la rúbrica (vease 5).
 - una carpeta con las evidencias que sean necesarias para mostrar que efectivamente se realizaron todos los cálculos mostrados en las métricas.

Los entregables deben ser enviados cada semana, únicamente por uno de los integrantes del grupo, bajo el siguiente formato:

CA_TP_NomApelAlumno1_NomApelAlumno2_NomApelAlumno3.zip

Nota: No se aceptarán entregas fuera del horario establecido.

5. Rúbrica de Calificación

Cuadro 1: Rúbrica del Trabajo Parcial. Los resultados deben incluir las métricas y sus valores, así como discusión de los mismo.)

Concepto	Puntaje
Introducción	1.0
Estado del Arte	1.5
Metodología	2.5
Experimentos	4.0
Resultados	5.0
Conclusiones	3.0
Presentación	3.0