

LENGUAJES DE PROGRAMACIÓN I
TRABAJO PRÁCTICO FINAL: ARIMAA
PRIMER SEMESTRE – AÑO 2020

Profesores:

Vicente González

Linda Riquelme

OBJETIVO

Desarrollar una aplicación en lenguaje C con interfaz gráfica GTK del juego ARIMAA. El mismo deberá permitir a un humano jugar contra la computadora y a dos programas distintos jugar entre sí. Debe tener una estrategia de juego, distinta a la simplemente aleatoria, por parte de la computadora la cual será considerada en la evaluación.

Los alumnos que no entregan el trabajo práctico final no se encuentran habilitados a rendir el examen final en ninguna de las oportunidades. Cualquier aclaración sobre estas especificaciones se podrá realizar a los profesores y las respuestas formarán parte de este documento como especificaciones del sistema. Las respuestas se publicarán por el AulaVirtual.

INTEGRANTES

El trabajo práctico se podrá realizar como máximo entre 2 (dos) personas.

ENTREGAS

PRIMERA ENTREGA

Fecha de entrega: Jueves 16 de abril de 2020

Documentación: Estructuras de datos a ser utilizadas, describiendo cada una de ellas y justificando la elección de cada estructura.

Juego: juego funcionando (con su código fuente correspondiente) de forma local en una terminal (sin interfaz gráfica). Debe funcionar el juego modo Humano vs. Computadora, cumpliendo con las reglas y especificaciones indicadas en la sección REGLAS DEL JUEGO de este documento. Se debe permitir al usuario elegir el color que usará cada jugador y ubicar sus diferentes fichas en el tablero. Ambas opciones también tienen la posibilidad de ser aleatorias. El juego debe terminar según las reglas.

No es necesario implementar aún la inteligencia del juego por parte de la computadora para esta entrega. Pero la computadora deberá realizar jugadas válidas en su turno, las cuales pueden ser aleatorias.

Si bien no se pide aún interfaz gráfica, las jugadas realizadas por ambos jugadores deben ser claras para el usuario en todo momento.

SEGUNDA ENTREGA

Fecha de entrega: Jueves 21 de mayo de 2020

Juego: Tablero del juego implementado en la interfaz gráfica GTK, con las siguientes funciones ya operativas: menú para salir del programa, menú de ayuda (que indique cómo jugar el programa, opciones, etc.), menú acerca del juego que muestre una ventana con los nombres de los autores del programa, ventana con campos para configurar una partida, menú juego nuevo que reinicia el juego.

ENTREGA FINAL Y DEFENSA

Fecha de entrega: jueves 11 de junio de 2020

Entregar ejecutable/s del juego, código/s fuente/s documentado y documentación (ver sección DOCUMENTACIÓN) en un archivo comprimido con la siguiente denominación **TPF_<NroMatricula1>_<NroMatricula2>.zip**.

El programa debe cumplir con los siguientes requerimientos:

- Cumplir con todas las reglas y especificaciones indicadas en la sección REGLAS DEL JUEGO. No se acepta otro conjunto de reglas, ni adicionar reglas.
- Tener interfaz gráfica desarrollada con GTK (uso de botones, menús, mouse, limitando el uso del teclado a lo imprescindible).
- Tener una estrategia de juego para la Computadora. No se acepta como estrategia el juego aleatorio.
- Tener la opción humano vs. computadora.
- Tener la opción computadora vs. computadora (el protocolo de comunicación entre ambas computadoras será definido en otro documento y publicado en el Aula Virtual).
- Estadística:
 - En el formato a su elección, se guardará en un archivo los siguientes datos estadísticos de cada jugador: Nombre del jugador, cantidad total de partidas jugadas, cantidad total de partidas ganadas, cantidad total de partidas perdidas, cantidad de empates.
 - Eligiendo el ítem “Estadística” del menú, se deberá mostrar en pantalla el registro histórico de los jugadores que jugaron el juego en dicha computadora, ordenando los registros según la cantidad de partidas ganadas (se muestran primero aquellos jugadores que tienen más partidas ganadas).
 - Solo las partidas terminadas serán consideradas en el cálculo de la estadística.
 - Los datos estadísticos deben permanecer luego de cerrar el programa. Es decir, independientemente de cuántas veces se cierre y abra el programa, los datos estadísticos deben ser consistentes con todas las partidas jugadas.
- Tener un menú funcional al inicio y fácilmente accesible durante el juego, con los siguientes ítems:
 - Juego nuevo: Durante el juego debe ser posible inicializar una nueva partida sin terminar la actual. La partida no terminada no es considerada en la estadística.
 - Ayuda: Contiene las reglas de juego y muestra cómo utilizar el programa.
 - Estadística: muestra los datos estadísticos de los jugadores. Se almacenan los datos estadísticos mediante un archivo para que persista cuando se cierra el programa.
 - Información sobre los autores (Acerca de): nombre de los autores del programa, año de realización.
 - Salir: cierra el programa.
- Deberá permitir una configuración de una partida con las siguientes configuraciones:
 - Permitir ingresar nombre de los jugadores.

- Elegir el modo de juego: Humano vs. Computadora o Computadora vs. Computadora.
 - Permitir elegir el color de fichas de cada jugador. Tener la opción que se elija aleatoriamente quién inicia (para la versión Humano vs. Computadora).
 - Permitir ubicar las fichas correspondientes a cada jugador. Tener la opción de ubicar aleatoriamente las fichas del humano (para la versión Humano vs. Computadora).
 - Especificar un directorio compartido para el protocolo de comunicación en el modo Computadora vs. Computadora.
- Durante el juego se debe mostrar en pantalla:
 - El nombre del rival como el nombre de jugador propio (dejando en claro el color de ficha que corresponde a cada uno).
 - El tablero de juego actualizado.
 - Indicar el turno de quién es en todo momento.
 - Al finalizar el juego, mostrar en pantalla el desenlace del juego.

En cada entrega, cada uno de los integrantes del grupo defenderá el trabajo y será calificado de manera individual, de modo a evaluar la participación de cada alumno. **En caso de no presentarse a la defensa el día de la entrega final, el trabajo será considerado como no entregado por los alumnos.** Se podrán hacer preguntas a cualquiera de los integrantes del grupo sobre cualquier parte del código, el programa, etc.

DOCUMENTACIÓN

Deberá incluir:

- Descripción de las estructuras de datos utilizadas y las funciones. Para las funciones describir el objetivo de la función, los parámetros de entrada, la salida y/o retorno de la función y qué sucede con los parámetros en caso de ser modificados.
- Descripción y justificación de la inteligencia/estrategia utilizada.
- Puntos fuertes del programa.
- Puntos débiles del programa.
- Dificultades encontradas durante el trabajo.
- Guía de utilización del programa (orientado a usuarios finales, explicando cómo usar el juego, sus opciones, menús, como mínimo).

EVALUACIÓN

La evaluación inicial es el correcto funcionamiento del juego según las especificaciones del trabajo, es decir, se realiza una partida, esta partida termina indicando el ganador que corresponda. **Si esta primera evaluación no es superada, se aplicará una penalización del 50% del puntaje de la entrega.**

Se establecerán criterios de evaluación en base a este documento de especificaciones y a cada criterio los profesores le asignan un puntaje. Cada criterio es evaluado para ver si se cumple, se cumple parcialmente o no se cumple. El puntaje obtenido será proporcional (si cumple completamente el criterio obtiene el total de puntos, si cumple parcialmente obtiene una parte, etc.).

Sobre la base de esta evaluación se define el puntaje final obtenido. Un criterio muy importante será el conocimiento de cada uno de los integrantes del grupo del programa realizado, este criterio se verificará haciendo preguntas específicas a los alumnos sobre las funciones implementadas.

Las primeras dos revisiones tienen un valor de 20% del total del trabajo cada una. La revisión final tendrá una calificación de 60% del total. En esa última revisión se aplica el criterio de corrección indicado en el párrafo inicial de esta sección.

El 100% del trabajo corresponde a 25 puntos del total de 100 puntos del semestre.

PUNTOS BONO – EXTRAS (puntos fuera de escala)

Estos puntos serán otorgados sólo a los trabajos que cumplan con los requerimientos solicitados y funcionen correctamente. Solo podrá ser considerado si se implementaron ambas modalidades Humano vs Computadora y Computadora vs. Computadora.

Ganar competencia entre programas (3 puntos bono)

Se llevarán a cabo partidas de todos contra todos o de eliminación directa, de acuerdo a la cantidad de trabajos presentados. El programa con mayor cantidad de partidas ganadas o ganador de la final, se llevará los puntos bono. Para realizar la competencia deben haber al menos 2 grupos con modalidad Computadora vs Computadora funcionando correctamente.

Mejor interfaz gráfica (2 puntos bono)

El trabajo con la interfaz gráfica considerada mejor realizada a criterio de los profesores llevará los puntos bono. Algunas consideraciones incluyen el aspecto visual al ejecutar el programa, la facilidad de uso para un tercero que no programó, las funcionalidades del mismo, que sea un programa robusto sin terminar abruptamente o que responda a los comandos de mouse sin retardo.

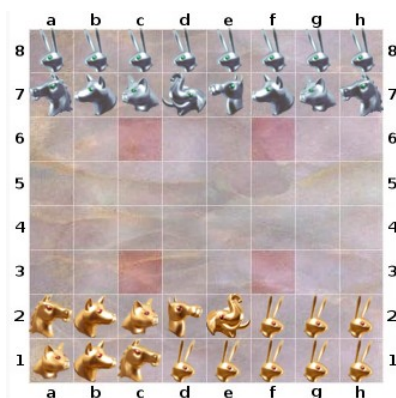
Los puntos bono se suman directamente al puntaje obtenido en el trabajo. Por ejemplo, si se obtuvo 25 puntos en el trabajo y además se realizaron dos puntos bono, entonces el puntaje obtenido será 27 puntos del total de 100 del semestre.

REGLAS DEL JUEGO: ARIMAA

Arimaa es un juego de mesa de estrategias que se juega entre dos jugadores. El tablero es de 8x8, donde 4 de sus celdas son celdas especiales denominadas “trampas”. Las piezas son de 6 tipos y representan a diferentes animales, algunos más débiles que otros más fuertes. Las piezas pueden ser capturadas cuando son ubicadas en las trampas, en caso de no tener piezas amigas (del mismo color) en las posiciones adyacentes.

A cada jugador, Oro y Plata, le corresponde 16 fichas: 1 elefante (🐘), 1 camello (🐪), 2 caballos (🐎), 2 perros (🐕), 2 gatos (🐈) y 8 conejos (🐰). Las fichas listadas anteriormente aparecen en orden del más fuerte al más débil.

El objetivo principal del juego es que cualquiera de los participantes logre ubicar a uno de sus conejos en la primera línea de su oponente (meta). Así, en el tablero de la figura a continuación, el jugador plateado ganará al ubicar uno de sus conejos en la fila 1, mientras que el oro podrá ganar si llega a ubicar a uno de sus conejos en la posición 8.



Tablero de Arimaa: cada jugador ubica sus fichas de la forma que desee o aleatoriamente.

Además, es posible capturar piezas del oponente al empujarlas o estirarlas hacia cualquiera de las celdas “trampa”. En el tablero anterior, las celdas trampa son las ubicadas en las posiciones c3, f3, c6 y f6.

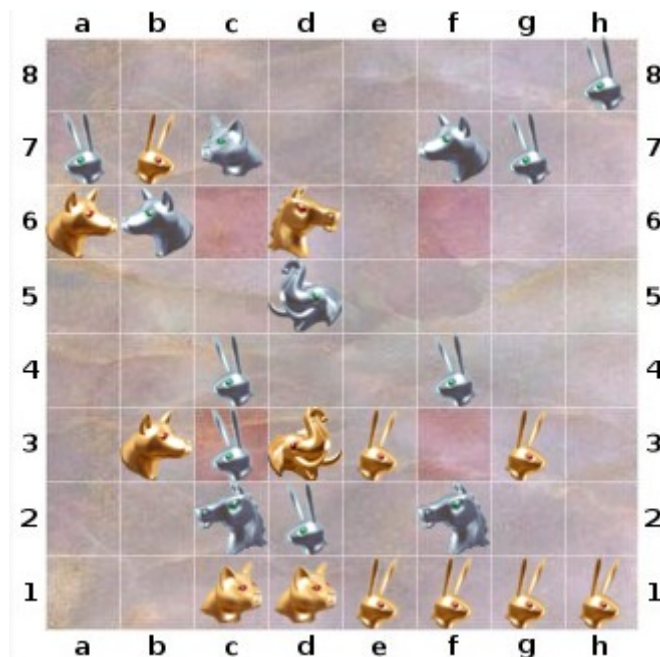
Al iniciar el juego, el jugador de oro ubica sus fichas en las filas 1 y 2 del tablero, mientras que el jugador plata ubica las suyas en las filas 7 y 8. Cada jugador puede decidir ubicar cada una de sus fichas en sus filas correspondientes sin ningún orden en particular. También se debe ofrecer la posibilidad de ubicar las fichas aleatoriamente. Dada esta configuración, para ganar, algún conejo dorado debe llegar hasta la fila 8. De la misma manera, si algún conejo plateado llega a la fila 1, el jugador plata gana el juego.

El juego debe permitir al usuario indicar quién inicia el juego. En cada turno, un jugador puede realizar hasta 4 movimientos (o pasar el turno luego de realizar, obligatoriamente, 1 movimiento). Cada movimiento consiste en mover una de sus piezas a una casilla adyacente (izquierda, derecha, arriba o abajo) vacía. Para el caso de los conejos, sin embargo, está prohibido el movimiento hacia abajo.

Una pieza puede empujar o estirar a una pieza enemiga más débil, que esté adyacente a la pieza atacante en cualquiera de las adyacencias (izquierda, derecha, arriba y abajo). Si el movimiento de una pieza implica el movimiento de más de una pieza enemiga (ya sea por empujarla o estirla), es necesario especificar la ficha enemiga que se desea mover.

Al estirar, la pieza más fuerte se mueve hacia una casilla adyacente vacía y la pieza más débil es estirada posicionándose en la casilla original del más fuerte. Por ejemplo, en la siguiente figura, el elefante en d5 tiene espacios adyacentes vacíos y puede estirar a una ficha más débil adyacente al moverse a una de dichas posiciones. Así, este puede pasar a la posición d4 (o c5 o e5) y estirar al caballo en d6 a la posición d5. Una pieza puede empujar a otra enemiga (siempre que esta última sea más débil) a una posición adyacente vacía; en cuyo caso la pieza más fuerte quedaría ocupando la posición anterior de la más débil. Así, por ejemplo, el

elefante dorado en d3 puede empujar al conejo plateado en d2 a la posición e2. Luego de hacerlo, el elefante pasaría a ocupar la posición anterior del conejo: d2. Una jugada de “estirar” o “empujar” cuenta como la realización de dos movimientos en ese turno.



Tablero Arimaa: movimientos de ataque

Las piezas que caigan en las celdas de trampa son consideradas como capturadas y son eliminadas del juego, a menos que ésta tenga alguna pieza del mismo color en las posiciones adyacentes. Por ejemplo, en la figura anterior, es posible capturar al caballo dorado en d6, empujándolo a c6 mediante el elefante plateado en d5. Además, las fichas no capturadas que se encuentren en celdas trampas pueden convertirse en capturadas cuando sus fichas del mismo color adyacentes se mueven y lo dejan sin protección. Este último sería el caso de que el caballo plateado en c2 y el conejo plateado en c4 se muevan y dejen sin protección al conejo en c3 (quien termina finalmente capturado y eliminado del tablero).

Finalmente, gana el juego quien logre ubicar a uno de sus conejos en su respectiva meta. En el caso particular de que un jugador se quede sin conejos (todos llegan a ser capturados), éste se queda sin posibilidades de ganar por lo que la partida debe finalizar y dar como ganador al jugador contrario. De manera a limitar el fin del juego, el programa debe permitir especificar la cantidad máxima de jugadas (turnos) permitidas. En caso de especificarse dicho valor, con cada cambio de turno se debe descontar la cantidad de jugadas restantes y, al quedarse sin jugadas disponibles, el juego deberá finalizar. En esas condiciones, el ganador será aquél que tiene la mayor cantidad de fichas en el tablero (o tenga una menor cantidad de fichas capturadas).