

**Instituto Tecnológico de Costa Rica (ITCR)**  
**Ingeniería en Computación, Alajuela (ICAL)**  
**Escuela de Computación**  
**Curso: Lenguajes de programación**

## **Tarea programada 04**

**Tema:** Programación imperativa lenguaje C

**Entrega:** Un archivo zip que incluya el código fuente de C y documentación en un archivo pdf **a través del TEC-digital**.

**Modo de trabajo:** Individual

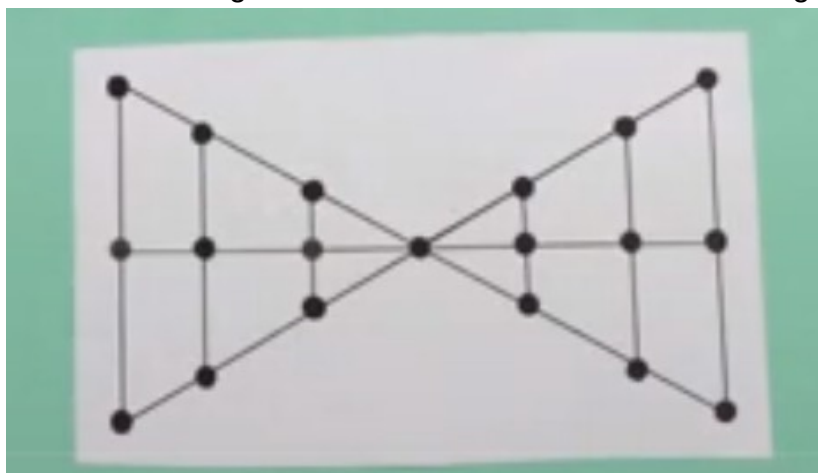
El objetivo del presente trabajo es poner en práctica el conocimiento adquirido durante el curso sobre programación imperativa y diseñar e implementar funcionalidad utilizando el lenguaje de programación C.

### **Descripción de la tarea:**

La tarea consiste en crear una aplicación con el **lenguaje de programación C** que juegue Lau-kata-kati. Lau-kata-kati es un juego parecido al de Damas chinas pero simplificado, donde dos oponentes se enfrentan en un tablero de juego con forma de mariposa con el objetivo de capturar fichas opuestas o bloquearlas para que no se muevan. Más información y un vídeo de cómo jugar está disponible en <https://www.whatdowedoallday.com/lau-kata-kati/>

Se requiere (Wikipedia<sup>1</sup>):

- Llevar el estado de dos jugadores.
- El tablero consiste de un grafo bidireccional como lo muestra la figura 1:



*Figura 1. Tablero del juego. Imagen de <https://www.whatdowedoallday.com/lau-kata-kati/>*

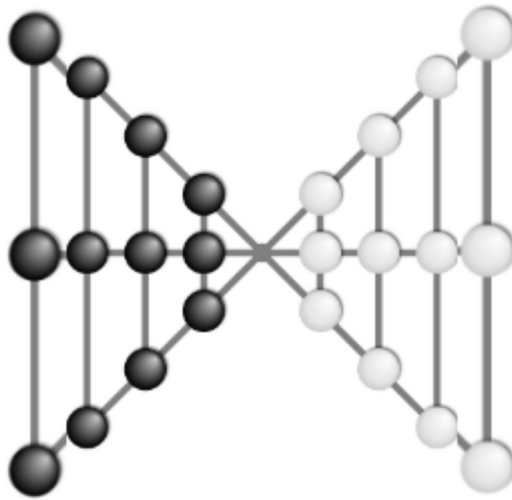
---

<sup>1</sup> [https://es.wikipedia.org/wiki/Lau\\_kata\\_kati](https://es.wikipedia.org/wiki/Lau_kata_kati)

- La profundidad de los triángulos puede variar. La figura 1 presenta dos triángulos de profundidad 3, es decir del punto central de la figura hacia la derecha o hacia la izquierda hay 3 filas con 3 nodos cada una. Un triángulo de profundidad 4 sería como el que muestra la figura 2.
- Cada jugador cuenta con 9 fichas, independientemente del tamaño de los triángulos.
- Las fichas de cada jugador se colocan en la base del triángulo siempre llenando las 3 filas de la base.
- Los jugadores pueden elegir quién mueve primero y la ubicación de sus fichas del lado izquierdo o derecho del grafo. Los jugadores alternan sus turnos utilizando una sola pieza para mover o capturar por turno.
- Una pieza se mueve una casilla por turno hacia un punto de intersección vacío adyacente siguiendo el modelo del tablero.
- Las capturas se realizan mediante un salto corto como en las Damas chinas, donde la pieza enemiga adyacente salta por encima de la del oponente hacia un punto vacante del otro lado. La captura se debe hacer en línea recta, siguiendo el modelo del tablero. Las capturas múltiples se permiten siempre y cuando haya un punto vacío entre las piezas enemigas, y un punto vacante más allá de la última pieza enemiga. En un solo movimiento se pueden capturar todas las fichas que cumplan con ese requisito.
- Si un jugador no puede realizar un movimiento o una captura porque sus piezas han sido bloqueadas o inmovilizadas por las piezas del otro jugador, esto se conoce como punto muerto, y el jugador pierde y gana el oponente.
- Si ningún jugador puede capturar más, el jugador con más piezas gana. Si ambos jugadores tienen la misma cantidad de piezas, entonces, el juego es un empate.

### **Funcionalidad a implementar:**

1. Crear una estructura de datos utilizando punteros que permita realizar los movimientos del juego (es importante tener en cuenta que las fichas se pueden mover en todas las direcciones por lo que las conexiones entre nodos deben ser bidireccionales). El usuario debe poder definir la profundidad de los triángulos que forman parte del tablero. La figura 1 presenta triángulos de profundidad 3. Un triángulo de profundidad 4 sería como el que muestra la figura 2.



*Figura 2. Tablero del juego con profundidad 4. Imagen modificada de Wikipedia*

2. Crear las estructuras que permitan implementar las reglas del juego.
3. Leer del teclado la ficha que cada jugador desea mover y realizar las actualizaciones en las estructuras de datos. Se debe leer la posición inicial y final de la ficha a mover.
4. Desplegar el estado del juego de forma amigable (no tiene que ser gráfica).
5. Implementar las reglas del juego de forma que un jugador no pueda realizar un movimiento inapropiado ya sea sin capturar o capturando una o más fichas del oponente.
6. Implementar funcionalidad que verifique cuándo un jugador ganó.

#### Requerimientos administrativos:

- La tarea debe programarse en el **Lenguaje C utilizando punteros** (la tarea no será aceptada en otro lenguaje de programación o implementada con otra forma de manejo de memoria).
- Se debe entregar **documentación** de la tarea en formato digital que incluya una descripción del sistema, diseño de componentes (estructuras de datos utilizadas, mantenimiento del estado del juego, jugadores, entre otros) y cómo ejecutar el código para jugar. Se debe incluir pantallas de captura para una secuencia de juego.
- El código debe estar documentado internamente. Debe contar con un encabezado al inicio del código que describe el juego y cómo se inicia a jugar. Además, para cada función se debe describir en detalle lo que hace, sus entradas y salidas.

## Rúbrica

	Puntos
<b>Documentación</b>	
Presenta descripción del sistema	1
Describe los componentes internos de la implementación del juego y cómo jugar.	5
Se incluyen capturas de pantallas representativas de una corrida.	5
La redacción y ortografía es muy buena en todas las secciones del documento. El documento está escrito en tercera persona y no tiene más de tres faltas ortográficas o de redacción.	3
<b>Implementación</b>	
Se crea una estructura de datos basada en punteros que permite realizar los movimientos del juego utilizando conexiones bidireccionales. El usuario debe poder definir la profundidad de los triángulos que forman parte del tablero por lo que se debe utilizar memoria dinámica.	10
El código está documentado apropiadamente (como se describe en este documento).	4
Se implementa interfaz agradable para verificar el estado del tablero.	3
Se crean las estructuras de datos adicionales al tablero que permiten implementar las reglas del juego.	1
Se lee del teclado la ficha que cada jugador desea mover y se realizan las actualizaciones en las estructuras de datos. Se debe leer la posición inicial y final de la ficha a mover. Se implementan las reglas del juego de forma que un jugador no pueda realizar un movimiento inapropiado ya sea sin capturar o capturando una o más fichas del oponente.	10
Se implementa funcionalidad que verifique cuándo un jugador ganó.	2