

# Laboratorio 2: Sistemas Operativos

**Profesor:** Viktor Tapia  
**Ayudante de cátedra:** Juan Pablo Varas  
**Ayudante de Tarea:** Javiera Cárdenas

03 de Abril 2023

## 1 Reglas Generales

Para la siguiente tarea se debe realizar un código programado en lenguaje C. Se exigirá que los archivos se presenten de la forma más limpia y legible posible. Deberá incluir un archivo README con las instrucciones de uso de sus programas junto a cualquier indicación que sea necesaria, y un archivo MAKE para poder ejecutar el programa.

## 2 Tarea

Se solicita recrear el juego de mesa llamado Toma 6, para esto se les pide que desarrollen el juego y que además de que un participante pueda jugar a través de la consola, se creen 3 jugadores extra para que el juego quede de 4 jugadores en total. Los jugadores extras deben ser creados con la función `fork()`.

### 2.1 Conceptos básicos

Se especificarán los conceptos básicos, cualquier cambio a estos debe ser especificado en el README que entreguen junto con el programa:

1. Se tienen 44 cartas en total, en donde cada una representa un número entero del 1 al 44, además cada una de estas cartas tiene una cantidad de puntos asociada a ella:
  - (a) Cartas múltiplos de 5: 2 puntos.
  - (b) Cartas múltiplos de 10: 3 puntos.
  - (c) Cartas múltiplos de 11: 5 puntos.
  - (d) El resto de las cartas tiene un valor de 1 punto.
2. El concepto clave del juego es la colocación de las cartas. En la mesa tendremos una serie de 4 filas, compuestas por cartas. Cada carta debe ser colocada en aquella fila cuya última carta de la fila sea inferior a la carta a colocar, y que, además, la diferencia entre esa última carta de la fila y la carta a colocar sea la menor posible.

### 2.2 Inicio de la partida

- Se desordena todo el mazo de cartas.
- Se roban las 4 primeras cartas y se colocan al inicio de cada fila.
- Se reparte a cada jugador 10 cartas.

## 2.3 Desarrollo de la Partida

La partida dura solo una ronda con 10 turnos. Cada turno se desarrolla de la siguiente manera:

1. El jugador elige una de las cartas de su mano por consola, el resto de los jugadores deberán elegir la carta de menor valor de su mano que sea mayor a una carta de la ultima posición de una fila aleatoria, en caso de no tener cartas que cumplan esa regla, entonces deberán elegir la menor carta de su mano.
2. Las cartas elegidas se ordenan de menor a mayor y se van colocando sobre las filas (comprobando si algún jugador debe tomar cartas de alguna fila).
3. En el momento que la carta de un jugador se coloque en la sexta posición de una fila, ese jugador deberá tomar las cinco cartas anteriores a una pila de descarte, anotando en su marcador el número puntos correspondientes a la suma de esas cartas, y dejando como inicio de la fila la carta que eligió al inicio del turno.
4. Si la carta de un jugador es inferior a todas las cartas finales de cada fila, entonces el jugador deberá elegir una fila y tomar todas las cartas que la componen a una pila de descarte, anotando los puntos correspondientes, dejando como comienzo de la fila la carta que eligió al inicio del turno, en el caso de las computadoras estas deben elegir una fila aleatoria.
5. Una vez que se han colocado todas las cartas, se repite el proceso hasta que todos los jugadores se queden sin cartas.
6. En el caso de que la ultima carta de la fila sea 44, esta solo puede ser cambiada con el punto 4 explicado en esta sección.

## 2.4 Fin de la Partida

Al finalizar la partida se cuentan los puntos de cada jugador y gana el jugador que tenga menos puntos.

## 3 Jugadores y procesos

El juego se compone de 4 jugadores (siendo usted uno de ellos). Cada jugador debe ser un proceso distinto, por lo que en total su programa estará ejecutando 5 procesos (un padre y 4 hijos). La acción que tendrá cada jugador, será escoger la carta a jugar y el conteo de sus puntos. Cada jugador comenzará con 10 cartas y según avance el juego, se deben ir eliminando de la mano. El orden de los jugadores en cada turno debe ser definido por el jugador muestre la menor carta entre todos los jugadores. Una vez que gana un jugador, el proceso padre debe indicar cuál de estos fue el que ganó, notificar a los procesos hijos, y luego se deben cerrar todos los procesos. Recuerde que para finalizar el programa, primero deben finalizar los procesos hijos, y luego el proceso padre.

### 3.1 Visualización de la Consola

Queda a su creatividad diseñar el formato de visualización del juego. Como mínimo se requiere que se muestren los siguientes elementos:

- Mano del Jugador: se deben poder saber las cartas del jugador principal en consola y debe permitir escoger la carta a jugar en su turno. El resto de los jugadores deben ser automáticos bajo las reglas indicadas anteriormente.
- Turno Actual: Debe estar indicado por pantalla el jugador que esté jugando. Un pequeño mensaje bastará.
- Últimas carta de cada fila: Cada vez que inicie un turno se debe mostrar la ultima carta de cada fila junto con la cantidad de cartas que presenta esa fila.

## 4 Presentación Aleatoria

Para cada tarea, se seleccionarán grupos al azar para presentar su tarea frente a ayudantes y eventualmente profesor, recibiendo una ponderación del 80% y 20% entre tarea y presentación respectivamente. Si su grupo presentó en una tarea, no volverá a salir nuevamente. Se comunicará días antes que grupos presentarán.

## 5 README

Debe contener como mínimo:

- Nombre, Rol y Paralelo de los integrantes.
- Especificación de los nombres de los archivos.
- Instrucciones generales de compilación y uso.

## 6 Consideraciones Generales

- Se deberá trabajar de a pares. Se deberá entregar en Aula a mas tardar el día 30 de Abril de 2023 a las 23:59 horas. Se descontarán 5 puntos por cada hora o fracción de atraso. Las copias serán evaluadas con nota 0 en el promedio de las tareas.
- La tarea debe ser hecha en el lenguaje C. Se asume que usted sabe programar en este lenguaje, ha tenido vivencias con el, o que aprende con rapidez.
- Pueden crear todas las funciones auxiliares que deseen, siempre y cuando estén debidamente comentadas.
- Las tareas serán ejecutadas en **Linux**, cualquier tarea que no se pueda ejecutar en dicho sistema operativo, partirá de nota máxima 60.
- Los archivos deberán ser comprimidos y enviados juntos en un archivo .tar.gz en el formato **TAREA2\_ROL1\_ROL2**.
- Las preguntas deben ser hechas por Aula. De esta forma los demás grupos pueden beneficiarse en base a la pregunta, **se responderán consultas hasta 48 hrs. antes de la fecha y hora de entrega.**

- Si no se entrega README o MAKE, o si su programa no funciona, la nota es 0 hasta la corrección.
- Se descontarán 50 puntos por:
  - Mala implementación del Makefile.
  - No respetar el formato de entrega.