



---

## Tarea programada 1: Juego Uno y medio

---

### Descripción

El juego de Uno y medio está diseñado para ser la competencia directa del popular juego de cartas llamado Uno. Es un juego basado en turnos que está conformado por una serie de cartas de colores. Para este caso existirán cartas numeradas de 0 a 7 (inclusive) y cartas especiales.

Por cada carta numerada del 0 al 7 existirán 4 posibles colores: **rojo**, **verde**, **azul** y **naranja**. Adicionalmente por cada color existirán 2 cartas. De forma que existirán 64 cartas numéricas.

En cuanto a las cartas especiales habrán: *come 2*, *cancelar* y *buscar dentro de la pila de cartas*. Existirán tres cartas por cada tipo.

Las reglas de juego son las siguientes:

Existirá una pila de cartas barajadas previamente boca abajo que será accesible para todos los jugadores. Se repartirán 5 cartas a cada uno, se tomará una carta de la pila y se pondrá boca arriba. Seguidamente los jugadores intentarán poner cartas válidas sobre la que se encuentra en el tope de la pila. Una carta es válida si tiene el mismo color que la carta que se encuentra en el tope o si es el mismo número. En el caso de las cartas especiales, se pueden utilizar en cualquier momento, representan cualquier color, por lo que la persona que sigue podrá poner la carta que guste luego.

En cuanto a las cartas especiales si un jugador utiliza el *come 2*, el siguiente deberá comer dos cartas de la pila a menos que tenga en su poder un *come 2* o un *cancelar* o un *buscar en la pila*. En caso de utilizar un *come 2*, el siguiente jugador deberá comer 4 cartas a menos que tenga un *cancelar* o un *come 2* y así sucesivamente.

La carta *buscar dentro de la pila de cartas*, permitirá a un usuario buscar entre las cartas que han sido jugadas y elegir una para jugar. El uso de esta carta no contará como parte del turno, por lo que el jugador podrá: jugar la carta especial o buscar una carta y jugar la carta que buscó en la pila.

Si un jugador no tiene una carta que poner, deberá comer de la pila de cartas, finalizará su turno y continuará del siguiente jugador hacia la derecha.

El juego termina cuando un jugador queda sin cartas. En caso de que se acaben las cartas del mazo, se tomará la pila de cartas, revolverá y volverá a utilizar como cartas elegibles.

El juego permitirá escoger dos opciones de juego: un humano contra otro (en formato *hot seat*) un humano contra la computadora. Al finalizar el juego, se presentará a los usuarios la opción de jugar de nuevo o salir.



## Entregables

Se deberá realizar un análisis y diseño del problema. Se espera que en un documento se especifique el problema, los pasos para llevar a cabo la solución, un diseño de clases (en formato UML) y una explicación de los algoritmos complejos utilizando pseudocódigo. Un plan de pruebas y los resultados de las pruebas.

Adicionalmente agregar una sección de problemas encontrados a la hora de resolver la tarea y un porcentaje de distribución del 100% del trabajo de cada uno de los integrantes. Por ejemplo si el grupo es de dos personas y cada estudiante trabajo la misma cantidad, el porcentaje será 50% cada uno.

El código fuente deberá contar con documentación interna y deberá generar la documentación utilizando Doxygen.

Generar un manual de usuario con instrucciones para el uso del juego.

Deberá programar el juego para que sea completamente funcional y utilizar todas las técnicas de programación que considere necesarias y entregar el código fuente.

Adicionalmente, agregar el nombre de su repositorio que utilizó el Github. Se espera que utilice el sistema de control de versiones para agregar frecuentemente los cambios que está haciendo mediante el uso de commits y push al servidor de Github.

Se evaluará el uso de la herramienta de control de versiones usando la herramienta Git.

## Notas adicionales

- Se recomienda utilizar [www.draw.io](http://www.draw.io) para la generación del UML por facilidad de uso.
- Se debe crear una cuenta en *Github* utilizando el correo dado en clase.
- Se recomienda descargar *Sourcetree* o *Git* para llevar a cabo los *commits*.
- Se recomienda usar la estructura con *branches*.
- Recuerde que no se debe *commit* a master código inestable.
- Esta tarea se puede realizar en parejas.
- Deberá generar la documentación externa usando *Doxygen*.

## Desglose

- Documentación interna y externa 25% (documentación interna del código fuente usando Doxygen y los demás puntos explicados anteriormente para la elaboración de la documentación).
- Codificación de la solución 75% (incluye uso del sistema de control de versiones, solución propuesta, funcionalidad, buen diseño, apego a los estándares de código, detección de errores, efectividad y eficiencia de las estructuras de datos y los algoritmos).