

***Proyecto Programado I – Juegos de Solitario***

***Introducción***

Un juego de solitario es aquel que puede ser jugado por una sola persona. Hay juegos de solitario que se juegan utilizando fichas o piedras, mientras que otros se juegan con cartas. La mayoría de este tipo de juegos consiste en un rompecabezas que, dependiendo de su posición inicial, puede o no ser resuelto de una forma diferente cada vez.

Se desconoce el origen concreto de los juegos de solitario con cartas, pero se supone que se originaron con la invención de los naipes, a los cuales se les atribuye origen chino, siglo 9. La primera colección de juegos de solitario con cartas fue escrita por Lady Adelaide Cadogan en un libro ilustrado publicado en 1870.

***Solitario Golf***

Entre los diferentes tipos de juegos de solitario con cartas se encuentra el solitario Golf. Este es un juego donde el jugador intenta obtener la menor cantidad de puntos (al igual que el deporte Golf) durante el curso de nueve rondas (nueve hoyos). Es un juego que requiere de habilidad más que de suerte, contrario a la mayoría de los juegos de solitario. Se implementará la versión fácil del juego que permite pasar de K a A y viceversa.

El solitario Golf se juega con un naipe normal de 52 cartas barajado. Se ponen 35 cartas en 7 pilas de 5 cartas cara arriba y el resto de las cartas apiladas cara abajo, aparte (el mazo). Se utiliza también una pila para las cartas que se van descartando, inicialmente tiene sólo la primer carta del mazo.

Reglas:

- Una jugada consiste en descartar continuamente una serie de cartas de las columnas.
- Sólo las cartas que se encuentran en el tope de cada columna se pueden descartar.
- Cuando la carta en el tope de una columna es removida, la carta siguiente puede ser utilizada en la jugada.
- Las cartas se pueden descartar si su valor es mayor o menor (por uno) que el de la carta en el tope de la pila de descartadas, sin importar el palo.
- El orden de los valores es A, 2, 3, 4, 5, 6, 7, 8, 9, 10, J, Q, K. Se continúa con A después de K, y viceversa.
- Cuando no hay posibles jugadas, se voltea la primera carta del mazo y se pone en la pila de descartadas para continuar con otra jugada.
- No se pueden reubicar las cartas de las columnas, sólo descartarlas.
- El juego puede finalizar de varias formas:
  - Se acaban las cartas del mazo y no hay más movidas disponibles. El programa debe detectar si ya no quedan movidas disponibles y comunicárselo al usuario.
  - El usuario elimina todas las cartas cara arriba.

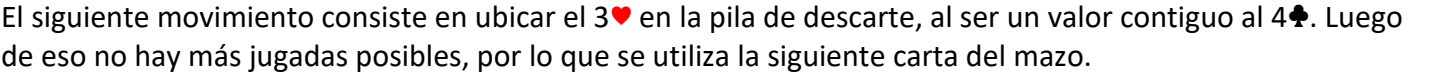
En la siguiente imagen se puede observar un ejemplo de la configuración inicial.



En este ejemplo, el 5♠ se movió a la pila de descarte porque el 5 es contiguo al 4♦.



La siguiente jugada consiste en mover el 4♣ a la pila de descarte porque es contiguo al 5♠.



Puntaje:

- El puntaje se calcula al finalizar una ronda.
- El jugador recibe un punto por cada carta que queda cara arriba.
- Si no quedan cartas, entonces recibe un punto negativo por cada carta que queda en el mazo.
- Después de las nueve rondas, un total de 45 puntos o menos se considera un puntaje regular, y un puntaje de cero o menor es un juego perfecto.

## ***Solitario Spider***

El propósito de este juego es eliminar todas las cartas de la mesa organizándolas en escaleras completas (descendentes desde la K hasta el A). Se juega con dos naipes de 52 cartas. En este proyecto se implementará la versión donde todas las cartas son del mismo palo.

Inicialmente, se reparten en diez pilas, las primeras cuatro con seis cartas y las restantes seis con cinco cartas. Solo la carta que está en el tope de cada pila está cara arriba. El resto de las cartas quedan en el mazo, cara abajo.

Se pueden mover grupos de cartas siempre y cuando estén ordenados en escalera. El grupo de cartas se puede ubicar sobre una carta que tenga el valor inmediatamente mayor.

El jugador puede repartir cartas del mazo, si no tiene más movimientos disponibles. Se ubica una carta cara arriba sobre cada una de las pilas. Se puede repartir si todas las pilas tienen al menos una carta.

Si se llega a tener una pila vacía, entonces el jugador puede ubicar la escalera que mejor le convenga en ese espacio libre.

Cuando se forma una escalera completa, la misma desaparece del juego.

En el siguiente ejemplo se observa la configuración inicial del juego.



Se mueve el 10 en la primera pila y se ubica sobre la J en la última fila. Se voltea automáticamente la carta que estaba bajo el 10.





El siguiente movimiento consiste en mover el A en la columna 9, hacia el 2 en la columna 2. Revelando un nuevo A en la columna 9. En este punto no hay más movimientos.



Como no hay más movimientos disponibles, se reparten cartas del mazo. Una por cada columna.



Luego de varias jugadas, el estado de la partida es la siguiente.



En este momento se puede formar una escalera descendiente moviendo del grupo de cartas que inicia en 4 de la columna 10, hacia la columna 5.



Al completarse una escalera del mismo palo, entonces se desaparece del juego.



## ***Software a desarrollar***

Su trabajo consiste en implementar los dos juegos Solitario Golf y Solitario Spider en el lenguaje C++. Para esto, debe utilizar estructuras de datos lineales como listas, colas y pilas. La interfaz de usuario del programa puede realizarse mediante consola.

Al iniciar la aplicación el usuario puede elegir el tipo de solitario que quiere jugar. Seguidamente inicia la partida.

### **Solitario Golf**

En Solitario Golf, el usuario tiene dos opciones: jugar una ronda única o jugar el juego completo (nueve rondas). Si el usuario elige una ronda única, se jugará ésta y el juego finaliza. Pero si elige el juego completo, el programa debe llevar control y comunicar al usuario cuál ronda se encuentra jugando y cuál es el puntaje acumulado.

Al iniciar una ronda, el programa debe realizar las siguientes tareas:

- Barajar el naipes
- Repartir cartas en las 7 columnas de 5 cartas cada una
- Presentar en pantalla el estado del juego. Deben aparecer los valores de todas las cartas que se encuentran con la figura arriba y también debe indicar cuántas cartas se encuentran en el mazo y cuál es la carta que se encuentra de primera en la pila de descarte.

En este punto el programa debe presentar al usuario opciones para:

- Elegir columna para descartar carta del tope
- Voltear una carta del mazo
- Deshacer
- Abandonar

El usuario puede interactuar con el juego mediante el teclado. Si en algún momento se intenta una jugada no válida, el programa no debe permitirlo. El programa debe ser robusto, no debe caerse por entradas no válidas del usuario.

La opción de deshacer le permite al usuario regresar a un estado anterior del juego. Esto debe reflejarse en la posición de las cartas como en la cantidad de movidas. Si no hay jugadas para deshacer, la opción no debe aparecer. No hay límite para la cantidad de veces que puede usarse la opción de deshacer, por esto es necesario que el programe guarde en una estructura todos los estados anteriores del juego.

Existen tres condiciones para que la ronda termine:

- La pila con las cartas del mazo queda vacía. Si esto sucede entonces el juego todavía debe tener cartas en las seis columnas, por lo que debe contarse la cantidad de cartas que están en las columnas y sumar esa cantidad al puntaje total.
- Las 6 columnas de cartas con la figura arriba quedan vacías. Si esto sucede entonces el mazo todavía debe tener cartas disponibles, por lo que debe contarse la cantidad de cartas en el mazo y ese valor (negativo) es el puntaje obtenido en esa ronda. También debe sumarse al puntaje total.
- El usuario elige la opción de abandonar.

Al finalizar cada ronda debe mostrarse el puntaje obtenido e indicar que se va a iniciar la siguiente ronda. Al finalizar el juego debe mostrarse el puntaje final indicando el tipo de puntaje que se obtuvo en las nueve rondas. Al finalizar el juego se retorna al menú principal.

## **Solitario Spider**

Si elige Solitario Spider, sólo se juega una partida, no hay rondas ni puntaje. El juego termina cuando el jugador logra eliminar todas las cartas de la mesa o si el usuario decide abandonar la partida. El programa no tiene que detectar si ya no tiene más jugadas posibles. Pero no debe permitir al usuario hacer jugadas inválidas.

Al iniciar la partida, el programa debe realizar las siguientes tareas:

- Barajar el mazo (son dos naipes completos)
- Repartir cartas en 10 columnas tal como se describió anteriormente
- Se presenta en pantalla claramente los valores de las cartas que se encuentran en el tope de cada columna

En este punto el usuario tiene opciones para mover una escalera y para repartir.

El programa debe detectar cualquier escalera completa, eliminarla de la mesa e indicar al usuario que ha completado una escalera.

Existen dos condiciones para terminar el juego:

- El jugador elimina todas las cartas de la mesa haciendo las ocho escaleras completas.
- El jugador utiliza una opción para abandonar el juego.

Al finalizar el juego, debe retornar al menú principal.



## ***Recomendaciones para la implementación***

Es importante que se abstraigan las diferentes funcionalidades que se necesitan en el juego, para programarlas separadamente y poder reutilizarlas.

Preferiblemente, aplique la técnica de separación de la lógica y la interfaz. Desarrolle clases controladoras que se encarguen de la lógica de cada juego y de exponer métodos con todas funcionalidades que están a disposición el usuario.

Analice estas funcionalidades e identifique qué entradas, salidas y restricciones tienen cada una de ellas. Identifique otras funcionalidades que le puedan ayudar a cumplir el objetivo de programar los juegos.

## ***Documentación***

Como documentación interna. Cada clase debe tener como documentación el autor o autores de la clase y para qué se utiliza la clase. Evite hacer comentarios excesivos. Escriba código claro y conciso, trate de apegarse a los principios de código limpio para el código que escriba. Recuerde que es mucho más importante que su código sea claro y fácil de entender que si es muy eficiente.

En cuanto a la documentación externa, debe entregarse dos documentos en formato PDF:

1. Manual de usuario: sección que explica detalladamente a cualquier usuario cómo ejecutar y utilizar la aplicación. Debe tener portada.
2. Documentación del proyecto
  - a. Portada. Incluye institución, escuela, curso, nombre del proyecto, profesor, integrantes y fecha.
  - b. Resumen ejecutivo. Descripción breve de todo lo que abarca la documentación. El objetivo de este resumen es captar la atención del lector y motivarlo a aprender más sobre el proyecto. Menos de una página.
  - c. Introducción. ¿Por qué se hace el proyecto y qué se incluye? (1-2 págs.)
  - d. Presentación y análisis del problema (5+ págs.)
    - i. Qué es lo que hay que resolver. Identificar pequeños problemas que deben resolverse en el proyecto.
    - ii. Cómo se va a resolver el problema. La forma en que se planea resolver el problema.
    - iii. Análisis crítico de la implementación. Luego de la implementación, decir qué se logró implementar, lo que faltó y qué cosas se podrían mejorar de lo que se implementó. No sólo mencionar, si no explicar por qué.
  - e. Conclusiones: resoluciones puntuales tras el proyecto. Estas deben ser relacionadas con los aspectos técnicos del trabajo únicamente. (1-2 págs.)
  - f. Recomendaciones: consejos o advertencias que se derivan de las conclusiones. Lecciones aprendidas durante el desarrollo del proyecto. Recomendaciones para personas que tengan que hacer el mismo trabajo. También deben estar orientadas con aspectos técnicos de la tarea programada. Se recomienda hacer una o más recomendaciones por cada conclusión. (1-2 págs.)
  - g. Referencias. Deben incluirse en formato APA.

## ***Forma de trabajo***

El proyecto se desarrollará en parejas o individualmente.

## ***Entrega***

Se hace por medio del TEC-Digital. Debe entregarse un archivo .ZIP con el código del proyecto y un PDF con la documentación.

## ***Evaluación***

La tarea tiene un valor de 20% de la nota final, en el rubro de Proyectos Programados.

Desglose de la evaluación de la tarea programada:

Documentación: 35%

Programación: 65%

## ***Recomendaciones adicionales***

Pruebe cada funcionalidad individualmente. No implemente todo el programa sin verificar el funcionamiento por separado de cada una de sus partes. Esto dirige a errores que son más difíciles de encontrar.

Recuerde que el trabajo es en equipos, es indispensable la comunicación y la coordinación entre los miembros del subgrupo.

Comparta el conocimiento con los demás compañeros de grupo y de la carrera, la ciencia de la computación es una disciplina que requiere el traspaso libre de conocimientos. Se logran mejores resultados con la colaboración de todos que con el esfuerzo separado de diferentes personas.

No dude en consultar diferentes fuentes para satisfacer las dudas. Aparte de las búsquedas en internet, asegúrese de exponer sus dudas a sus compañeros, profesor y conocidos que estudien la carrera; en la mayoría de las ocasiones es más provechosa conversación de 10 minutos entre personas que están trabajando en lo mismo que pasar horas buscando la respuesta a una duda de forma individual.

No deje la documentación para el final, es buena práctica ir desarrollándola durante todo el transcurso del proyecto. Recuerde que la documentación debe ser concisa y puntual, por lo que en realidad no toma mucho tiempo al realizarla de esta forma.

Plagios no serán tolerados bajo ninguna circunstancia. Cualquier intento de fraude será evaluado con una nota de cero y se enviará una carta al expediente del estudiante. Siempre escriba su propio código.

## ***Referencias***

Golf (patience). (2013, May 30). In Wikipedia, The Free Encyclopedia. Retrieved 23:32, September 17, 2013, from [http://en.wikipedia.org/w/index.php?title=Golf\\_\(patience\)&oldid=557594572](http://en.wikipedia.org/w/index.php?title=Golf_(patience)&oldid=557594572)

Spider (solitaire). (2017, August 9). In *Wikipedia, The Free Encyclopedia*. Retrieved 17:56, September 18, 2017, from [https://en.wikipedia.org/w/index.php?title=Spider\\_\(solitaire\)&oldid=794765818](https://en.wikipedia.org/w/index.php?title=Spider_(solitaire)&oldid=794765818)

Solitaire. (2013, August 30). In Wikipedia, The Free Encyclopedia. Retrieved 23:32, September 17, 2013, from <http://en.wikipedia.org/w/index.php?title=Solitaire&oldid=570748700>