

Universidad Simón Bolívar

Departamento de Computación y Tecnología de la Información

CI-2691 - Laboratorio de Algoritmos y Estructuras I

Trimestre Septiembre-Diciembre 2016

## **Proyecto: Solitaire Chess**

El objetivo de este proyecto es la implementación de un programa no trivial, haciendo uso del análisis descendente y de las técnicas vistas a lo largo del curso de teoría y práctica de Algoritmos y Estructuras I

### **Planteamiento del Problema**

Se quiere que usted implemente un juego llamado Solitaire Chess. Este es fundamentalmente un juego de lógica para un solo jugador y que usa las reglas del ajedrez de una manera simplificada para crear retos que deben resolverse.

### **Desarrollo y reglas de juego**

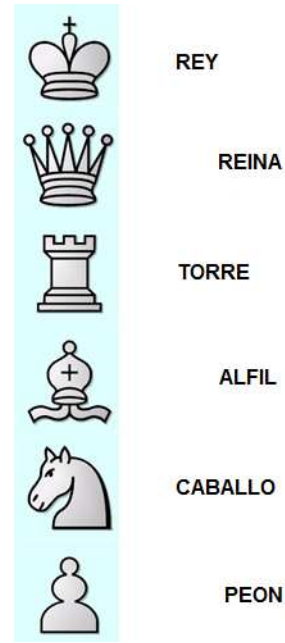
Solitaire Chess está inspirado en el juego del ajedrez y por lo tanto se hará una breve descripción de este juego. El ajedrez tiene su origen en la India, más concretamente en el Valle del Indo, y data del siglo VI d.C. Originalmente conocido como Chaturanga, o juego del ejército, se difundió rápidamente por las rutas comerciales, llegó a Persia, y desde allí al Imperio bizantino, extendiéndose posteriormente por toda Asia.

El ajedrez es un juego de dos jugadores que se enfrentan usando cada uno 16 piezas de un mismo color, “blancas” (piezas de color claro) o “negras” (piezas de color oscuro), las cuales se ubican sobre un tablero de 64 casillas (8 x 8) enumerado desde la columna “a” hasta la columna “h” y desde la fila 1 hasta la fila 8 (ver figura 1). El objetivo del juego es darle jaque mate al rey adversario (atacarlo con una pieza de tal modo que no pueda moverse a ninguna casilla que esté libre de ataque).

Todas las piezas pueden moverse respetando sus reglas de movimiento hasta una posición final, en la cual pueden capturar o comer otra pieza enemiga ubicada en dicha posición. Para lo anterior, se exceptúa el peón que sólo puede moverse hacia adelante y comer diagonalmente.



a) Tablero inicial del ajedrez

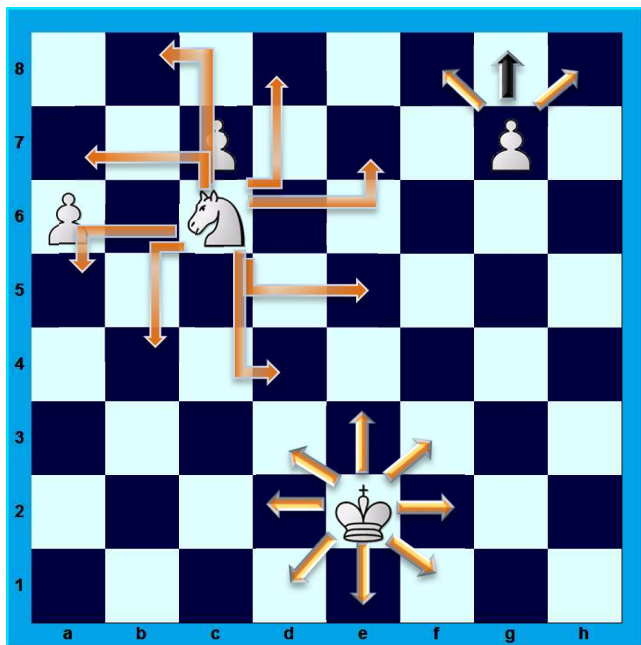


b) Piezas del ajedrez

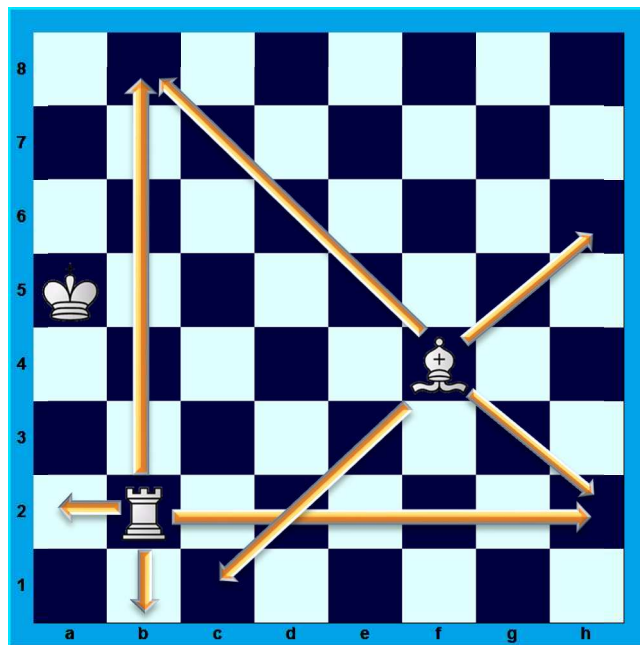
Figura 1: Tablero al comienzo del juego de ajedrez y sus piezas

A continuación, se describe el movimiento de cada pieza:

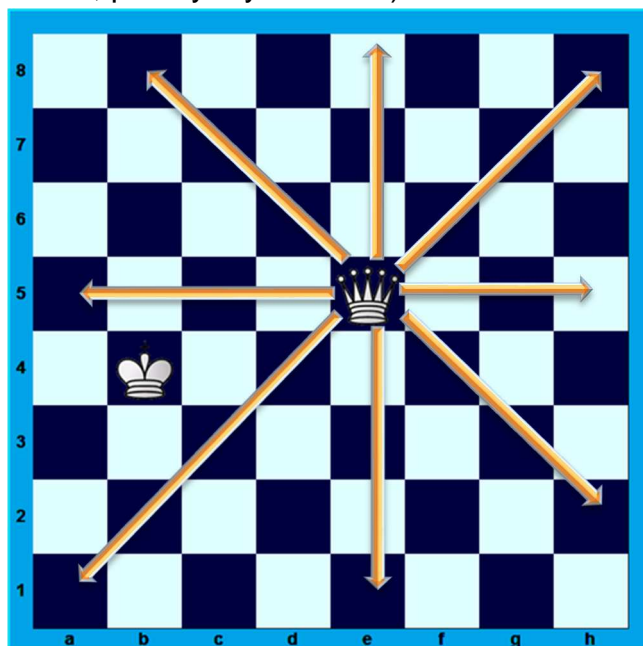
- Rey: puede moverse una sola casilla en cualquier dirección desde la posición actual. Por ejemplo, el rey puede moverse a cualquier casilla adyacente a la posición actual e2 en la cual está ubicado (ver figura 2a). Esta pieza no debe ser nunca capturada.
- Caballo: se mueve 3 casillas en forma de L desde la posición actual hasta la final. Por ejemplo, en la posición c6 (ver figura 2a) el caballo puede alcanzar las posiciones a5, b4, d4, etc., moviéndose en forma de L. Un ejemplo de L sería la secuencia de casillas siguiente: c6-b6-a6-a5. Note que a pesar de que en las posiciones a6 y c7 hay peones, el caballo puede “saltar” por encima de los mismos, siendo la única pieza que tiene esta capacidad.
- Peón: Solo puede desplazarse una casilla hacia adelante desde la posición actual. Por ejemplo, desde la posición g7 (ver figura 2a) sólo puede avanzar hasta la casilla g8, la cual debe estar desocupada y sólo puede comer diagonalmente si hay piezas ubicadas en las posiciones f8 y h8.



a) Movimientos del caballo, peón y rey



b) Movimientos de la torre y el alfil



c) Movimientos de la reina

Figura 2: Movimientos de las piezas del ajedrez

- Torre: puede moverse horizontal o verticalmente a cualquier casilla desde la posición actual. Por ejemplo, tal como se aprecia en la figura 2b, la torre ubicada en b2 puede trasladarse a cualquier casilla ubicada en la vertical b1-b2-b3-b4-b5-b6-b7-b8 o a cualquier casilla ubicada en la horizontal a2-b2-c2-d2-e2-f2-g2-h2, exceptuando obviamente la posición b2 en la cual ya está ubicada.
- Alfil: puede moverse diagonalmente a cualquier casilla desde la posición actual. Por ejemplo, desde la posición f4 (ver figura 2b) el alfil puede trasladarse a cualquier casilla ubicada en la diagonal b8-c7-d6-e5-f4-g3-h2 o a cualquier casilla ubicada en la diagonal c1-d2-e3-f4-g5-h6, exceptuando obviamente a f4.
- Reina: puede moverse en cualquier dirección desde la posición actual en forma diagonal, horizontal o vertical. Por ejemplo, la reina ubicada en e5 (ver figura 2c) puede avanzar hasta cualquier casilla de la diagonal, la horizontal y vertical que recorren todo el tablero y se interceptan en la posición e5.

## Descripción de Solitaire Chess

Solitario Chess es un rompecabezas de lógica e ingenio inventado por el finlandés Vesa Timonen y producido por ThinkFun desde 2010.

El juego es para un solo jugador y está basado en las reglas del ajedrez tradicional descrito anteriormente.

El objetivo es capturar todas las piezas del tablero, de acuerdo con las “cartas de desafío” y siguiendo las reglas de movimiento, hasta que sólo quede una sola pieza en el tablero. Hay algunas diferencias clave entre el Ajedrez clásico y el Solitaire Chess, pero los movimientos de las piezas siguen siendo los mismos:

- El ajedrez Solitaire es una versión simplificada con un cuarto del tamaño de un tablero de ajedrez normal (ver figura 3).
- Dado que el juego es para un solo jugador, todas las piezas son del mismo color y son sólo 10 piezas en total (ver figura 4).
- Una “carta de desafío” en el juego de mesa original “Solitaire Chess” es una tarjeta que tiene impreso una configuración del tablero en la cual no están ubicadas todas las piezas necesariamente (por ejemplo, ver figura 5).
- Cada vez que se realiza un movimiento, se debe capturar o comer una pieza. Por ejemplo, el movimiento de la figura 6 aunque es correcto para el caballo, sin embargo, no es válido porque no conduce a la captura de ninguna pieza. La figura 7, es un ejemplo de una jugada correcta. Cada desafío que incluye el Rey dará lugar a que sea la última pieza en el tablero, dada la restricción de que esta pieza no puede ser capturada por otra.

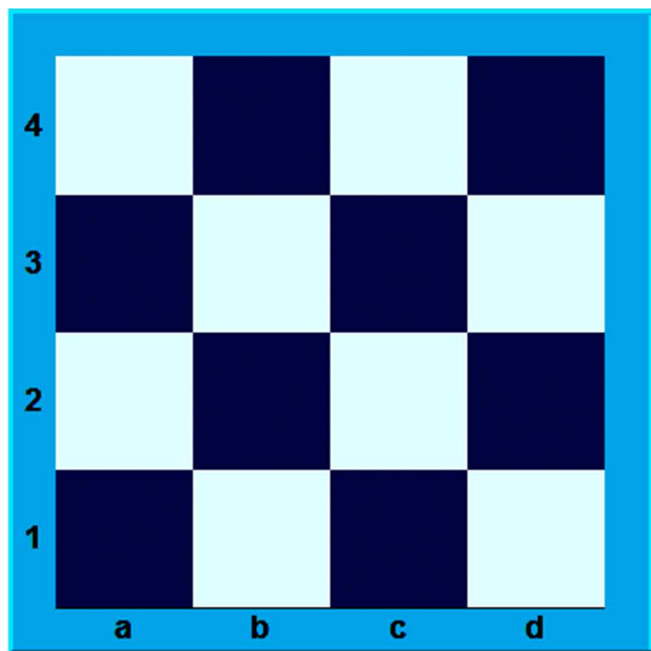


Figura 3: Tablero de Solitaire Chess

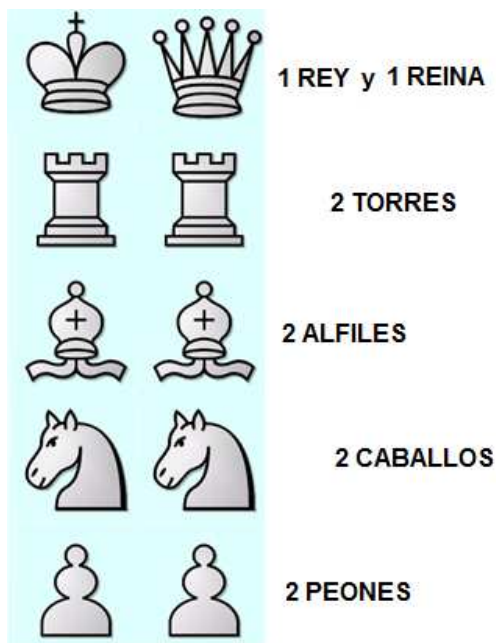


Figura 4: Tablero de Solitaire Chess

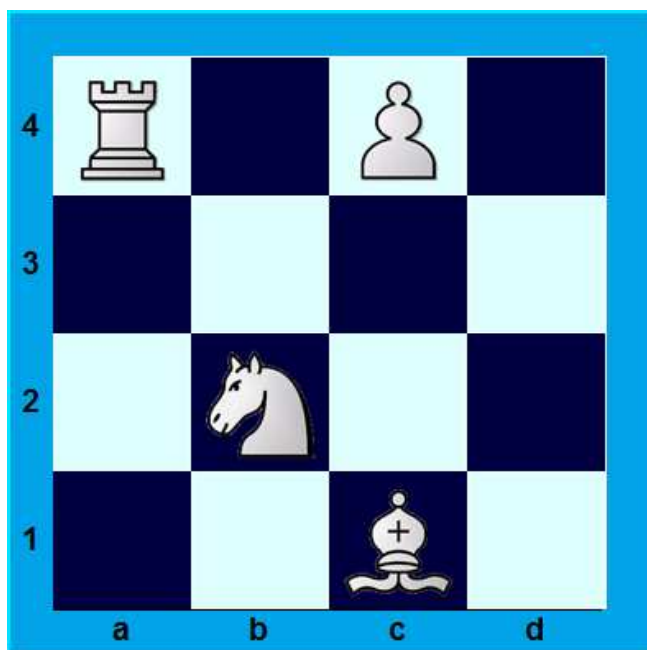


Figura 5: Ejemplo de configuración de Solitaire Chess

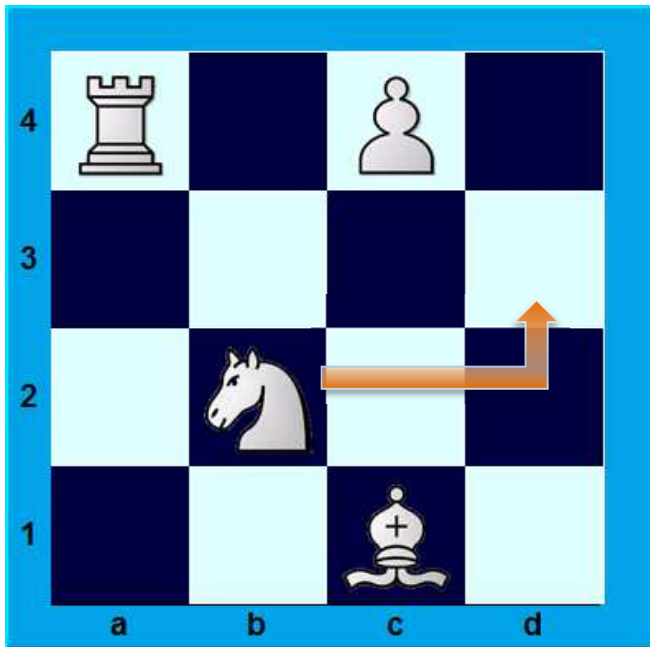


Figura 6: Ejemplo de jugada incorrecta

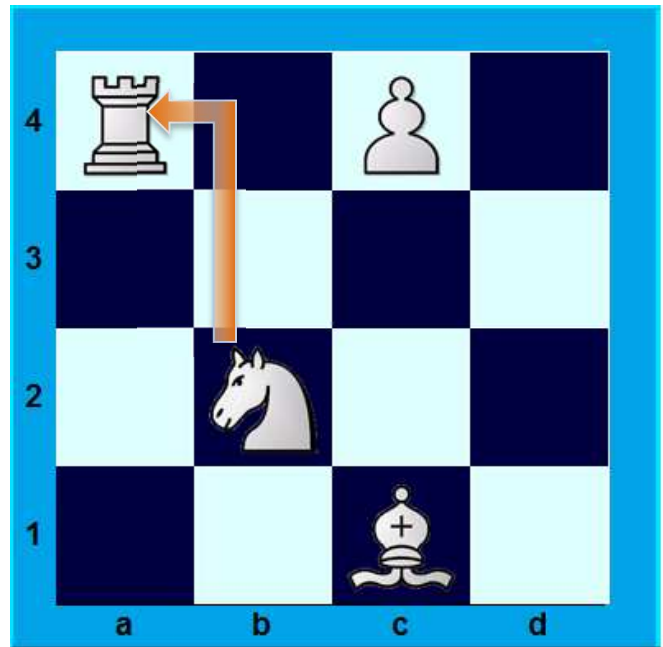
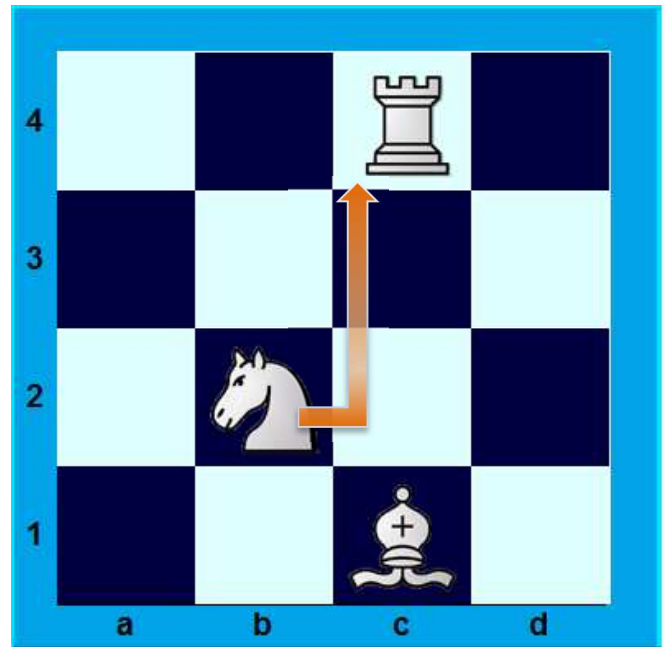
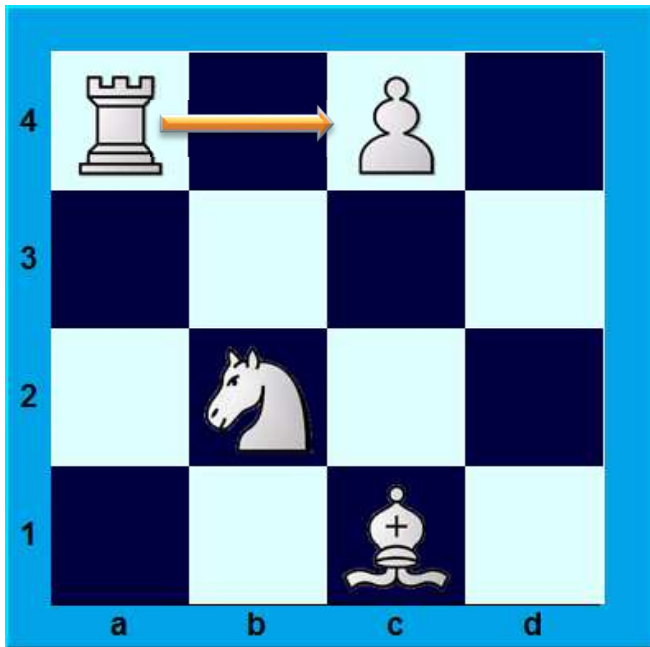


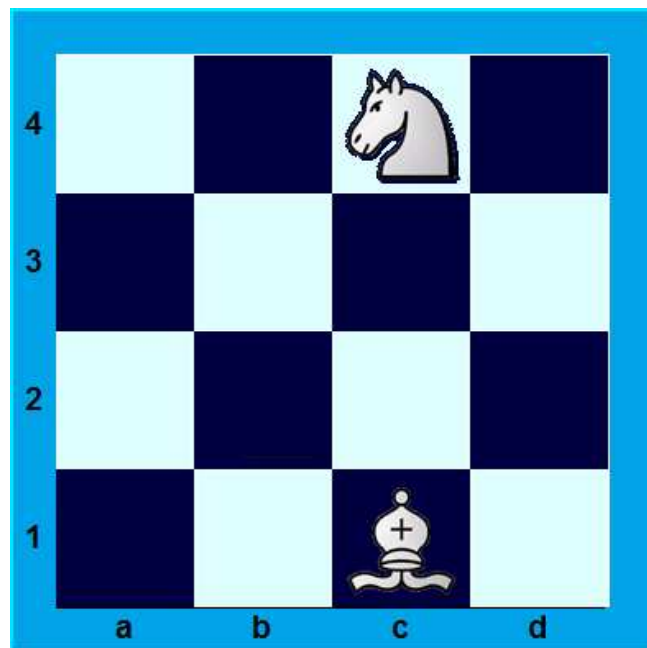
Figura 7: Ejemplo de jugada correcta

A continuación, podríamos resumir la dinámica del juego como sigue:

- Una vez que se carga una configuración inicial (“carta de desafío”), el jugador debe seleccionar una pieza con la cual procederá a capturar a otra pieza.
- Esta captura implica mover la pieza elegida a la posición de la pieza capturada, sustituyendo esta última.
- El procedimiento anterior debe seguirse reiteradas veces hasta que en el tablero ninguna pieza pueda ser capturada por alguna otra o hasta que quede una sola pieza.
- El primer caso se refiere a cuando en el tablero quedan dos o más piezas, ocasionando la pérdida de la partida (ver figura 8), mientras que en el segundo caso se gana la partida (ver figura 9)

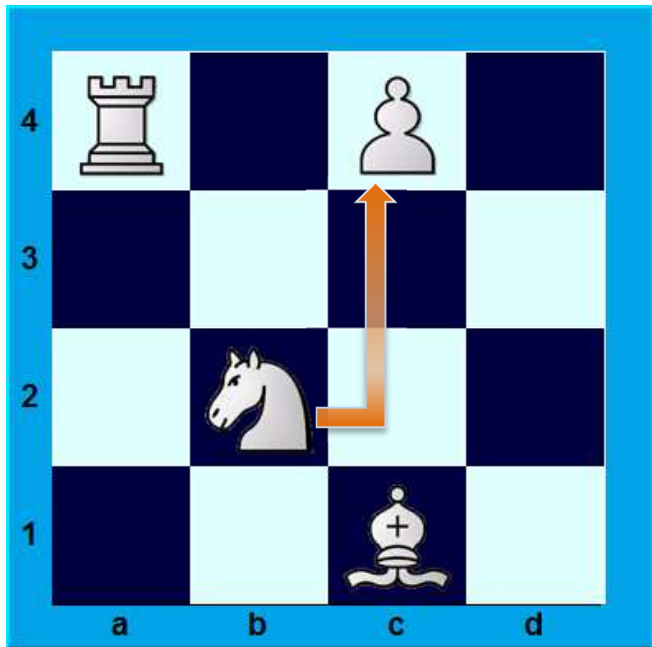


- a) Se elige mover la torre y se captura el peón      b) Se elige mover el caballo y se captura la torre

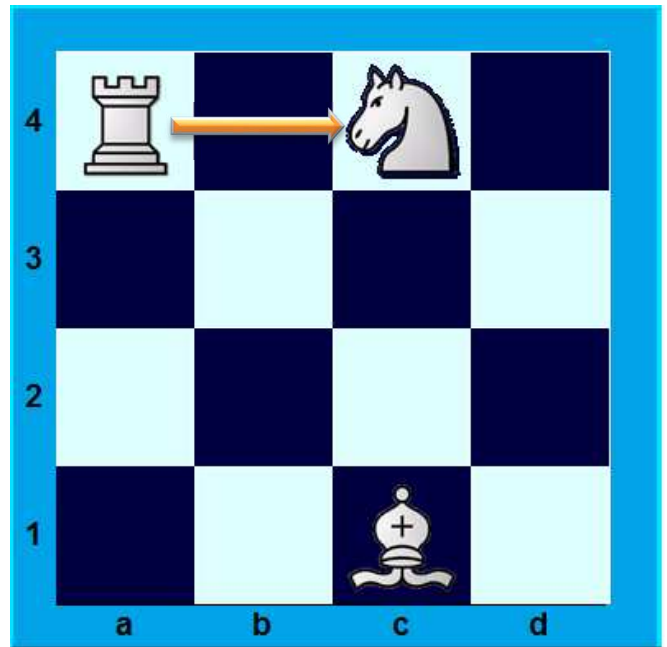


- c) Ninguna pieza puede ser capturada y se pierde la partida por quedar dos piezas en el tablero.

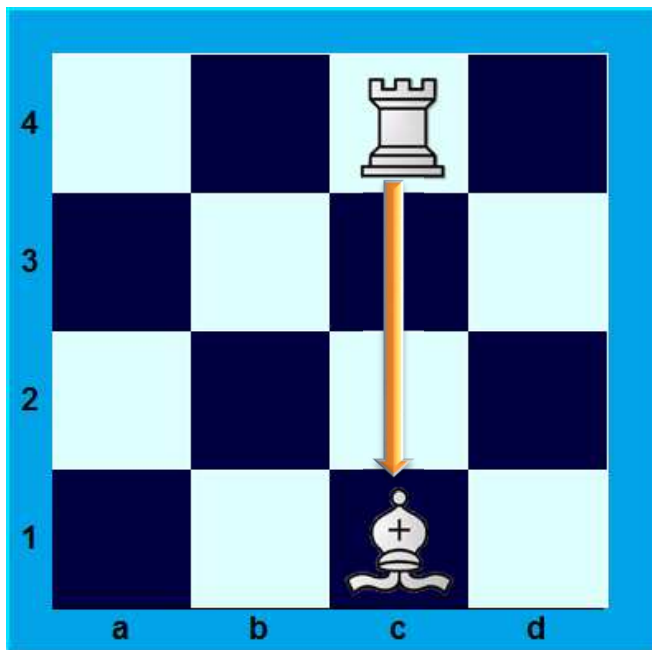
Figura 8: Ejemplo de secuencia de jugadas que provocan la pérdida de la partida



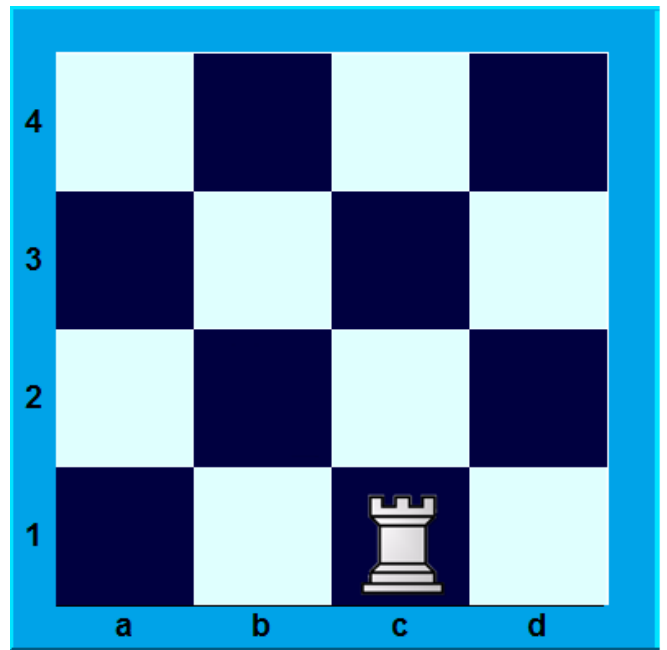
a) Se elige mover el caballo y se captura el peón



b) Se elige mover la torre y se captura el caballo



c) Se elige mover el caballo y se captura el peón



d) Se genera una configuración ganadora

Figura 9: Ejemplo de secuencia de jugadas que provocan ganar la partida



## Descripción de la implementación del juego a desarrollar

La versión de Solitaire Chess que debe implementarse cumple con las siguientes especificaciones:

Al cargar el juego en la pantalla inicial (antes de cargar el menú principal) el jugador introduce un nombre único que lo identificará mientras juega en la sesión actual. A continuación, se presenta el menú principal donde el jugador puede seleccionar una partida nueva, cargar una existente, mostrar la tabla de records, o salir del juego. En la figura 10 se puede apreciar un estilo de menú, sólo como referencia.

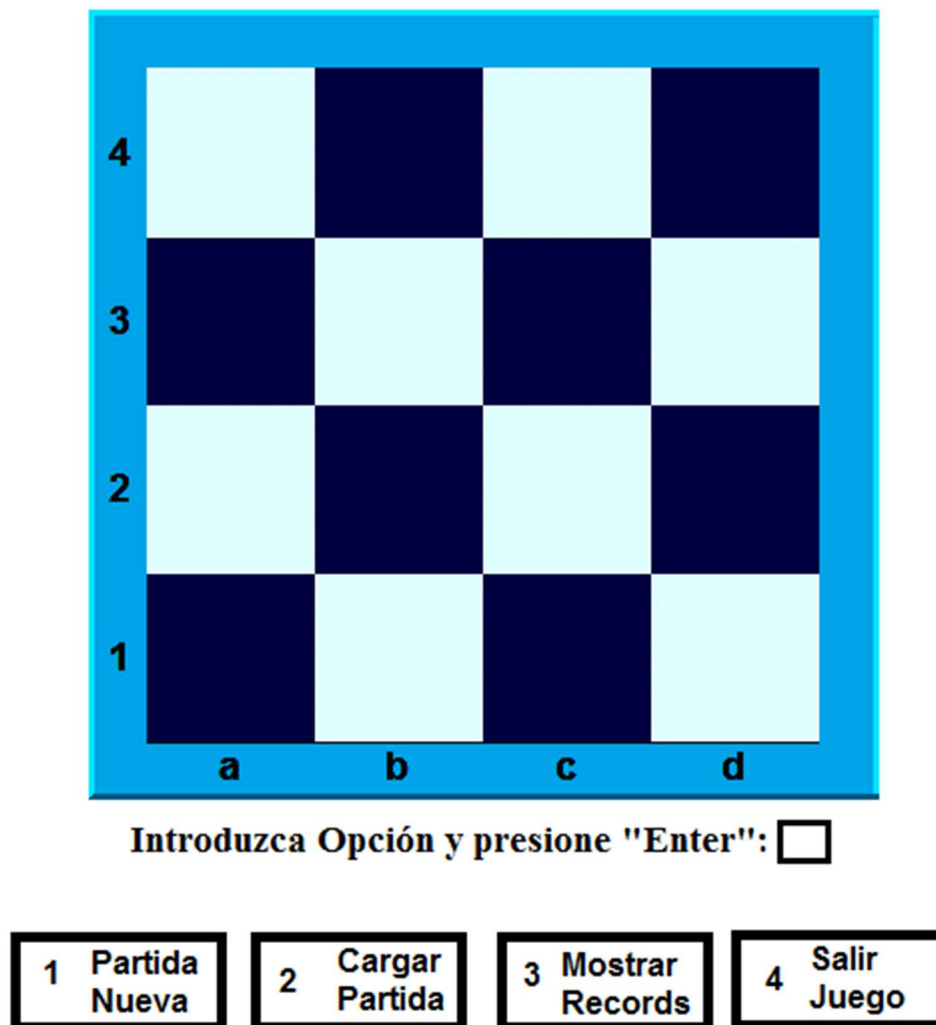


Figura 10: Ejemplo de menú principal de Solitaire Chess

Cuando el jugador selecciona una nueva partida, también debe seleccionar el nivel (fácil, difícil, muy difícil o entrenamiento) y el modo de carga para los niveles fácil, difícil o entrenamiento (al azar desde un archivo o desde el teclado); luego se presenta una pantalla al estilo de la figura 11. Cabe destacar que en el nivel fácil se disponen de 3 minutos por partida y la posibilidad de deshacer la última jugada realizada. En el nivel difícil se dispone de un minuto y 30 segundos por cada partida y no se pueden deshacer jugadas. En el nivel muy difícil se dispone de dos minutos y 3 partidas que deben ser resueltas una a continuación de la otra mientras el tiempo transcurre, sin posibilidad de hacer pausas ni retroceder jugadas. El nivel de entrenamiento dispone de todas las opciones del menú de la figura 11 (excepto el tiempo que no se usa) y adicionalmente una opción “SOLUCION”, la cual, dada la posición de una pieza inicial, permite obtener una posible solución a partir de dicha pieza, mostrándola en caso de existir o notificando que no existe solución.

Si se elige “Cargar Partida”, una vez indicado el número de la misma, la fecha y el nivel, debe restaurarse la partida en las condiciones exactas en las que fue guardada.

“Mostrar Records”, muestra en pantalla los records de partidas ganadas por nivel y en orden descendente por el número de victorias, seguido por el orden alfanumérico del nombre del usuario.

“Salir juego”, sale del juego en forma definitiva, antes de lo cual muestra al jugador un mensaje de confirmación.

Cada opción es seleccionada ingresando su número correspondiente y presionando “Enter”, como se aprecia en la figura 10.

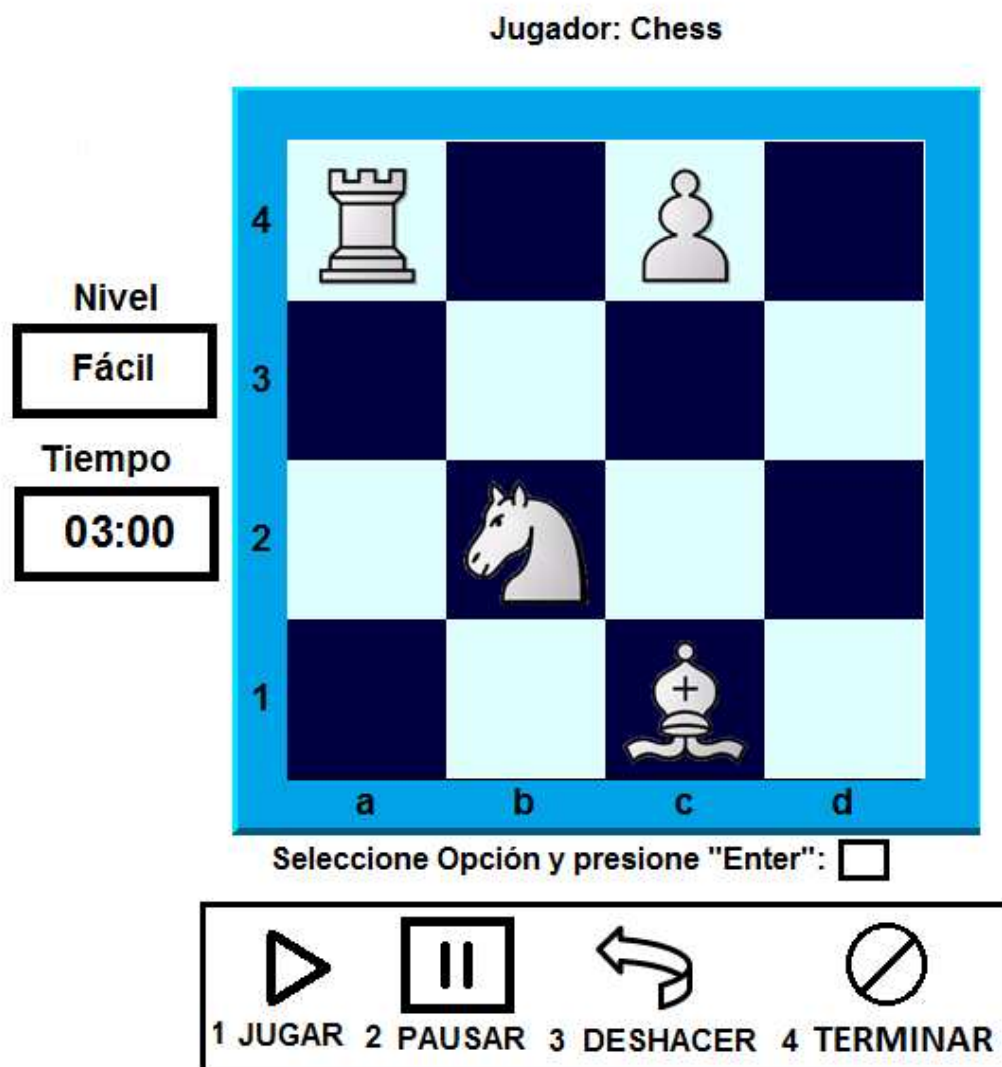


Figura 11: Ejemplo de menú de Partida Nueva

Cuando se elige una partida nueva ésta puede ser cargada al azar de un archivo ("partidasnuevas.txt") o directamente a través del teclado. Por cada partida nueva hay una nomenclatura asociada que describe las piezas y sus posiciones. Por ejemplo, para jugar una partida como en la figura 5 en el nivel "Fácil" debe estar registrada una entrada en el archivo "partidasnuevas.txt" o bien dicha entrada ingresarla por el teclado. El formato de la entrada asociada a una partida nueva es como sigue:

Ta4-c4-Cb2-Ac1 Fácil (en el caso de la posición c4 se sobreentiende que ahí se ubica un peón como en la notación usada en ajedrez)

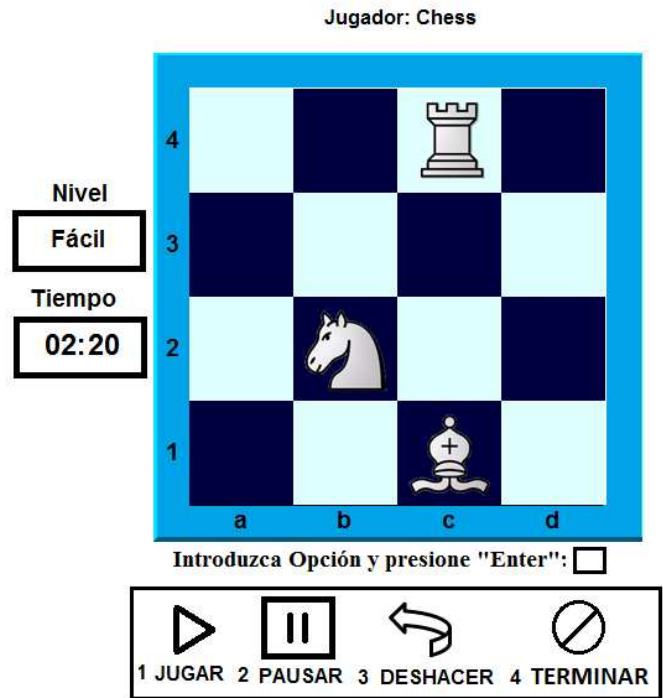
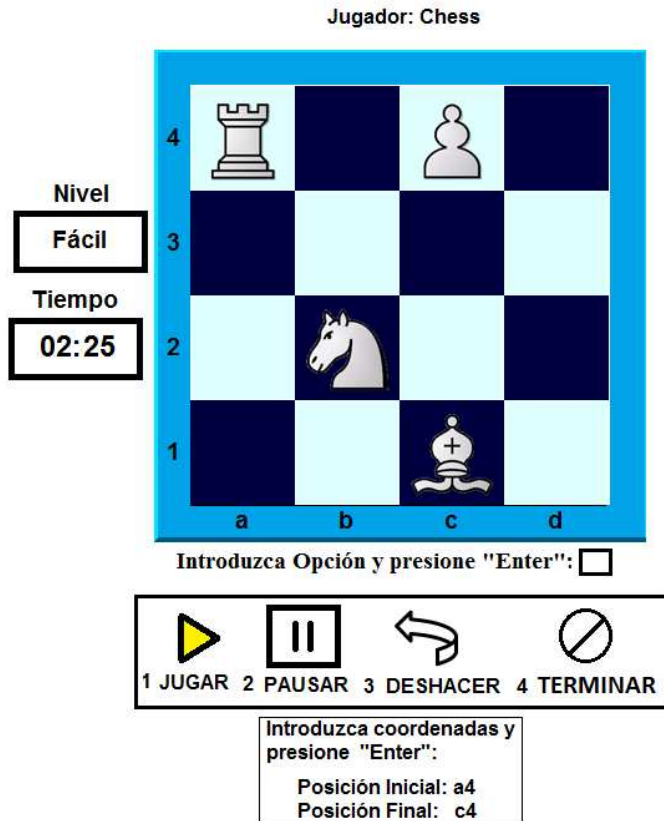
En el caso de que se elija una partida y una vez que el jugador elige “JUGAR”, el tiempo de juego se inicia en cuenta regresiva. Si al llegar el tiempo a 00:00 el jugador no ha concluido el desafío entonces se pierde la partida y en caso contrario se gana, debiéndose en ambos casos notificar al jugador mediante un mensaje en pantalla.

Cuando el jugador gana, se debe verificar el archivo “records.txt” y se debe guardar en la posición correspondiente el record obtenido, pudiéndose presentar dos casos: que al jugador nunca se le ha registrado un record, en cuyo caso se guarda en la posición correspondiente su nombre, número de victorias (una victoria en este caso) y el nivel o si el jugador ya tiene records registrados para ese nivel se le debe incrementar en uno el número de victorias registradas. En la figura 12 se puede apreciar un ejemplo de ejecución de una jugada al introducir la opción 1.

Cuando el jugador elige “PAUSAR”, se coloca el tablero en blanco y el tiempo se detiene de inmediato en el valor actual. Cuando el jugador vuelve a elegir la opción “JUGAR” se restablece la configuración que tenía el tablero y se reanuda la cuenta regresiva del tiempo.

Cuando se elige “DESHACER”, el tablero retorna a la configuración inmediatamente anterior a la actual, excepto el tiempo que sigue corriendo. Además, se asume que al menos se ha realizado una jugada y el juego no está en pausa ni en el nivel difícil, en el cual no se permite deshacer jugadas.

Cuando el jugador elige “TERMINAR”, antes de devolver el juego a la pantalla principal (figura 10), debe preguntársele si desea guardar la partida (sólo para los niveles fácil y difícil) y en caso afirmativo se debe guardar automáticamente una entrada para la partida actual en un archivo (partidasguardadas.txt) usando la siguiente nomenclatura: “Partida N fecha Tiempo Nivel ConfigTablero” , donde N es el número de partida, fecha es la fecha actual del sistema (día, mes y año), Tiempo es el tiempo transcurrido, Nivel es el nivel de juego y ConfigTablero se refiere a la nomenclatura antes explicada para una partida nueva. Por ejemplo, para la partida de la figura 11, si se decide guardar, la entrada en el archivo “partidasguardadas.txt” podría ser: Partida 5 21112016 Fácil Ta4-c4-Cb2-Ac1.



a) Se elige mover la torre para capturar el peón b) Al ser válida la jugada se ejecuta

Figura 12: Ejemplo de ejecución de una jugada a partir de tablero de la figura 11

## Estructura general del programa:

A continuación, se sugiere una estructura bastante general para dividir la funcionalidad de su programa:

- Una parte de carga e inicialización de componentes (tablero, reloj, botones).
- Inicializar partida: si es una partida guardada se carga del archivo de partidas guardadas y si es nueva del archivo de partidas nuevas o desde el teclado.
- Validar Opción Menú.
- Validar Evento Reloj.

- Validar jugada:
  - Validar jugada Dama
  - Validar jugada Rey
  - Validar jugada Torre
  - Validar jugada Alfil
  - Validar jugada Caballo
  - Validar jugada Peón
- Validar estado de pérdida (después de cada jugada): se debe validar que ninguna pieza pueda ser capturada por otra o dicho de otro modo que para cada pieza no se pueda acceder a la posición de las otras piezas que quedan sobre el tablero. En el caso de que el estado de pérdida se cumpla, se envía al jugador un mensaje indicando que ha perdido la partida y se finaliza la partida.

## **Especificaciones generales de la interfaz:**

- Deben indicarse en la pantalla todos los parámetros que correspondan (nombre del jugador, nivel, tiempo, etc.).
- Como prioridad el juego debe funcionar correctamente de acuerdo a todas las especificaciones de cada nivel. Aunque el diseño de interfaces no es el objetivo de este curso, debe recordar que implementará un juego gráfico (mínimo en 2D) y por lo tanto la interfaz debe presentar una mínima calidad en cuanto a colores, formas y efectos que logren una experiencia de juego “amigable” y “divertida”. Tenga en cuenta que una interfaz de calidad pudiera ser sencilla pero bien diseñada.
- La interfaz gráfica debe realizarse usando la librería gráfica de Python llamada Pygame que permite la creación de objetos en 2D.

## **Otras consideraciones:**

Cada función debe estar documentada con su nombre, parámetros de entrada y salida, breve descripción de lo que hace y las pre y post condiciones e invariantes y cota según aplique.

## Condiciones de la entrega

El proyecto se realizará por equipos de dos personas pertenecientes a la misma sección. El proyecto consta de dos entregas:

### Entrega 1 (8 puntos)

Implementación de las siguientes funcionalidades del juego:

- Menú principal con las opciones Salir Juego y Partida nueva (niveles fácil y difícil) y modo de carga desde el teclado.
- Menú de juego (niveles fácil y difícil) con las opciones “JUGAR”, “DESHACER” y “TERMINAR”. Recuerde que en el nivel difícil no se pueden deshacer las jugadas.

La primera entrega debe ser subida al aula virtual en un archivo llamado Entrega1ci2691SepDic16\_X\_Y.tgz, siendo X y Y los carnés de los integrantes del equipo, a más tardar a las 11:55 pm del viernes 02 de diciembre de 2016. Debe subirse al espacio del aula virtual de cada integrante.

Además, a más tardar el 05 de diciembre de 2016 debe entregar a su profesor de laboratorio la “Declaración de Autenticidad para Entregas” firmada por los integrantes del equipo.

### Entrega 2 (22 puntos)

Implementación del juego completo (14 puntos). La segunda entrega debe ser subida al aula virtual en un archivo llamado Entrega2ci2691SepDic16\_X\_Y.tgz, siendo X e Y los carnés de los integrantes del equipo, a más tardar para el viernes 16 de diciembre antes de las 7:30 am. Debe subirse al espacio del aula virtual de cada integrante.

Además, dos días antes, es decir, el miércoles 14 de diciembre de 2016 a las 8:00 am, se debe entregar a su profesor de laboratorio un sobre sellado (con el nombre, carné y profesor de laboratorio de cada integrante) el cual debe contener:

- La “Declaración de Autenticidad para Entregas” firmada por los integrantes del equipo.
- Un informe con la siguiente estructura (8 puntos):
  - Portada: Debe incluir los nombres y los números de carnés de los integrantes del equipo, así como el nombre del profesor encargado de cada integrante.
  - Introducción: Breve descripción del problema resuelto, cuáles son los objetivos planteados y el alcance de la solución. Debe indicar cuál es el contenido del informe.
  - Diseño: Explicación del análisis descendente hecho para resolver el problema. Descripción de las estructuras de datos utilizadas. Explicación de los subproblemas claves del programa.

- Estado actual del proyecto: Se debe indicar el grado de operatividad del programa, es decir, si funciona perfectamente o no; en caso negativo indique cuáles son los errores.
- Conclusiones: Resultados obtenidos, dificultades presentadas, su visión sobre la experiencia del desarrollo del proyecto y las recomendaciones que considere necesarias para mejorar los próximos cursos de algoritmos I.
- Bibliografía.

## Consideraciones finales

- Es obligatorio que el equipo trabaje de forma coordinada. El profesor podrá hacer un interrogatorio individual a cada integrante del equipo, y el no responder adecuadamente a las preguntas puede tener como consecuencias penalizaciones en la nota del proyecto.
- Cualquier error que sea hallado en este enunciado, así como cualquier tipo de observación adicional sobre el proyecto, serán publicadas como fe de erratas en la página web del curso.
- No debe haber copia, ni intercambio de información específica, ni ayuda detallada entre los equipos. El incurrir en cualquiera de las acciones descritas anteriormente tendrá como consecuencia sanciones severas.
- Si el código de alguna entrega no puede ser ejecutado por el interpretador, entonces la entrega será calificada con cero.
- El no cumplimiento de todos los requerimientos podrá resultar en el rechazo de su entrega.

## Referencias

- <http://www.thinkfun.com/products/solitaire-chess/>
- <http://www.puzzlesdeingenio.com/blog/analisis-review-del-juego-solitaire-chess-de-thinkfun/>