**INSTITUTO POLITECNICO NACIONAL**

**UPIIZ.**

**Análisis de Algoritmos**

**UNIDAD 2**

**Problema del caballo**

**Docente: Roberto Oswaldo Cruz Leija**

**Alumno: Cesar Delgado Padilla**

**3CM1**

**Marco Teórico**

Problema del caballo:

La ruta del caballo En ajedrez, un caballo es una pieza que se mueve en forma de L (un cuadro en uno de los ejes y dos cuadros sobre el otro). Una ruta de caballo en un tablero de ajedrez es una secuencia de movimientos echa por un caballo de tal forma que cada casilla del tablero es visitada una sola vez. Si la posición final del recorrido está a un salto de caballo de la posición inicial, la ruta se denomina reentrante. La siguiente figura muestra una ruta reentrante de caballo en un tablero de 8x8.



Aitor está intentando obtener otra ruta reentrante de caballo en un tablero de 8x8. Necesita construir un programa para verificar que la ruta realizada es correcta. Tu labor consistirá en construir el programa que necesita Aitor. Como parámetro de entrada tenemos un array de cadenas de texto, movimientos que contiene la secuencia de casillas visitadas por el caballo en orden. Cada elemento representa una celda única y sigue el siguiente patrón: donde es una letra que representa la columna y es un dígito que representa la fila. Por ejemplo, "A1" representa la esquina inferior izquierda del tablero, y “F6" representa la esquina superior derecha (ver figura). El programa debe devolver la cadena "Válido" si movimientos contiene una ruta reentrante de caballo e “Inválido” en caso contrario (todas las comillas se incluyen por claridad)

Consideraciones:

-evitar las casillas del centro

-no salirse del tablero

-marcar los movimientos

- saber Cuántos saltos dio

-número de filas

-número de columnas

Fila inicial

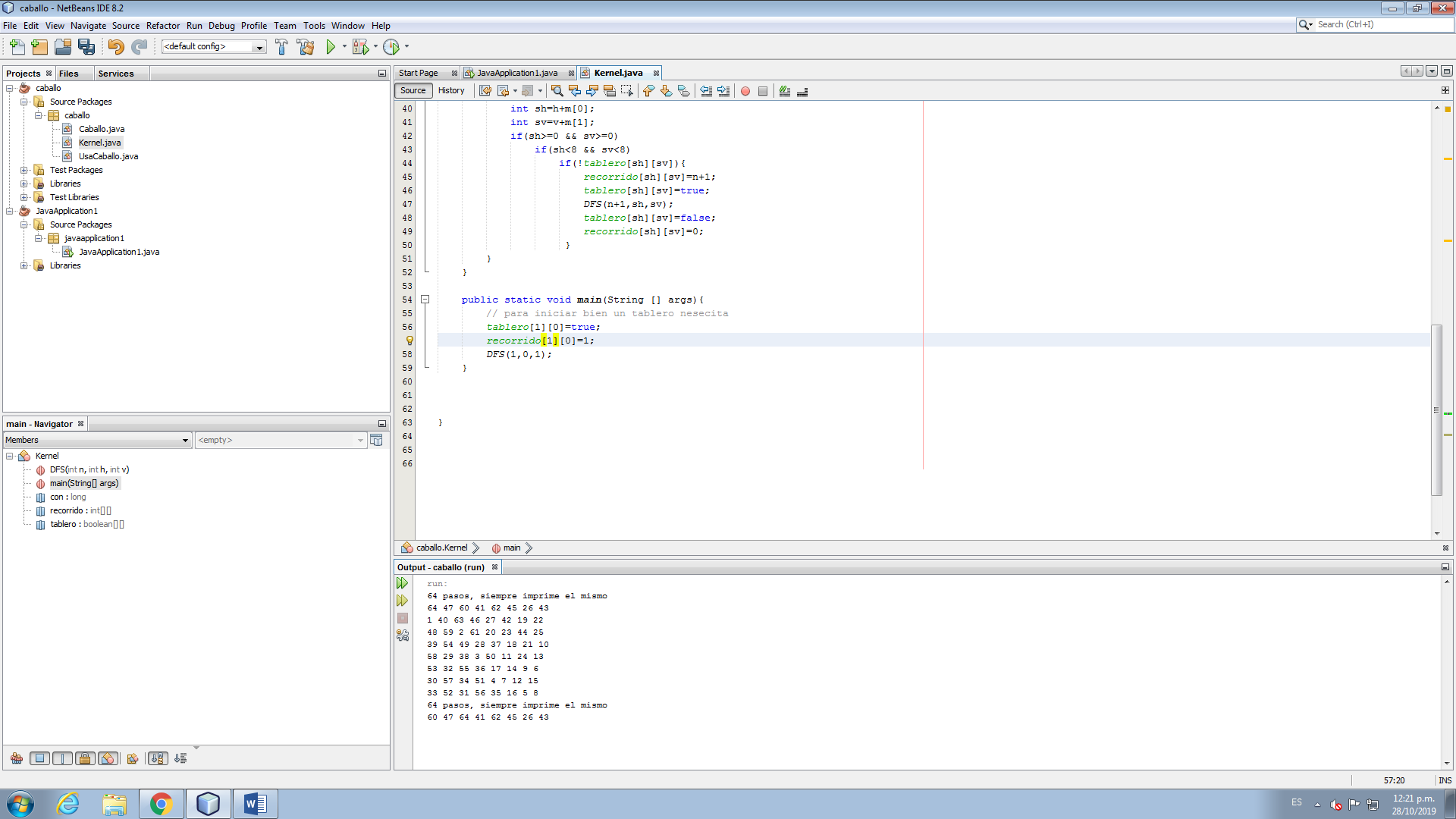
Columna inicial

**Restricciones**

- movimientos contendrá exactamente 64 elementos - Cada elemento de movimientos será de la forma "", donde es una letra en mayúscula entre la 'A' y la 'F', inclusive, y es un dígito entre '1' y '8', inclusive.

**PRUEBAS**

Prueba iniciando en (1,0)



Conclusiones:

En el problema del caballo nos dimos cuenta que al realizarlo con fuerza bruta, el algoritmo se hacía más tardado conforme incrementaba el tamaño de la matriz, cuando le dábamos la posición 1, 0 o 0,1 el algoritmo se tardaba un poco pero si lo realizaba, en cambio con otras posiciones tardaba un poco más, pero no nos dimos cuenta si en verdad lo terminaba ya que si tardaba muchísimo, el algoritmo consto de tres partes, el main que solo mandaba llamar las funciones para implementar el algoritmo, el dfs que es el que generaba el algoritmo y lo imprimía, y la parte que generaba el tablero en ceros.

Otra cosa que notamos es que si el algoritmo se hubiera realizado dinámico recursivo, al implementarlo se hubiera ejecutado muchísimo mas rápido.