

Primer Proyecto Programado -Paradigma Imperativo, lenguaje C

Objetivo:

Desarrollar una aplicación que permita aprovechar las particularidades de un lenguaje de programación del paradigma imperativo, en este caso C.

El caso: Alineamiento de fichas en un tablero

Se desea la construcción de un programa que permita la definición de un tablero de dimensión $n \times n$. Las fichas son posicionadas de manera aleatoria sobre las casillas, una ficha máximo por cuadro.

En una única movida, cualquier ficha puede ser trasladada a una posición no ocupada por otra ficha, de manera horizontal o vertical sobre el tablero.

La meta es alinear todas las n fichas en una línea recta ya sea horizontal, vertical o diagonalmente, utilizando la menor cantidad de movidas.

Ejemplo, para el caso donde $n = 5$ el tablero se muestra en la figura 1(a).

En la Figura 1(b) se muestra que con cinco movidas se alinea todas las fichas verticalmente. No existe un número menor de movidas que basten para crear una línea con cinco fichas.

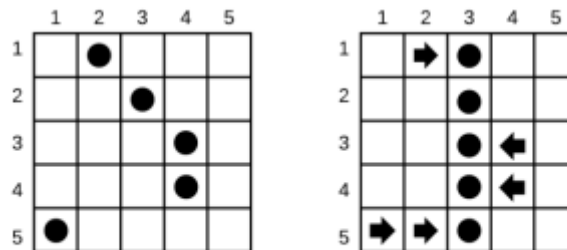


Figura 1. (a) Tablero inicial y (b) solución con 5 movidas para $n = 5$

Aunque existen otras soluciones usando seis movidas, por ejemplo, se puede alinear las fichas usando seis movidas en las columnas 2 y 4.

Datos de entrada

Los datos de entrada se recibirán por la línea de comandos, para lo que recibirá el nombre de un archivo de texto que contiene las configuraciones de los tableros que se desean probar.

El nombre del archivo de prueba será **PruebasGrupoN.txt** donde N es el número de grupo asignado para la elaboración de este proyecto.

El formato del archivo de texto debe respetar el siguiente orden de información:

El archivo puede contener varios casos a probar, cada uno de ellos se forma por dos líneas que definen la configuración del tablero a crear.

Primer Proyecto Programado -Paradigma Imperativo, lenguaje C

Cada caso inicia con una línea que contiene un número entero n , $1 \leq n \leq 15$. Este dato es la dimensión del tablero.

La línea siguiente contiene los números de fila y columna de la primera ficha, seguidos por los números y columna de la segunda, y así sucesivamente. Resulta fundamental la validación de que los datos contenidos en el archivo de entrada sean válidos. No necesariamente debe haber una ficha por fila o por columna.

Las filas y columnas están numeradas como en el diagrama anterior, es decir inician en 1.

Los datos de entrada para el último caso serán seguidos por una línea que contiene un único **cero**. Esto determina la finalización de la configuración para la prueba.

Ejemplo:

<p>Entrada de Ejemplo</p> <pre> 5 1 2 2 4 3 4 5 1 5 3 2 1 1 1 2 3 3 1 1 2 2 2 0 </pre>	<p>Se desea un tablero de 5x5. La siguiente línea coloca las fichas en las posiciones de la primera prueba.</p> <pre> 5 1 2 2 4 3 4 5 1 5 3 </pre> <p>El programa debería dar la respuesta para el primer escenario planteado.</p> <p>Luego se debe replantear el algoritmo para un tablero de dimensión 2x2, con su respectiva configuración.</p> <pre> 2 1 1 1 2 </pre> <p>El programa debería dar la respuesta para el segundo escenario planteado.</p> <p>Luego se debe replantear el algoritmo para un tablero de dimensión 3x3, con su respectiva configuración</p> <pre> 3 3 1 1 2 2 2 </pre> <p>El programa debería dar la respuesta para el segundo escenario planteado.</p>
<p>Al leer el 0 final, se entiende que es el final de las pruebas y se debe concluir el programa.</p>	

Para cada caso leído, el programa debe dibujar el tablero en pantalla tal y como fue configurado a partir de la lectura extraída del archivo de casos. La apariencia debe mostrarse en forma tabular tal y como aparece en el ejemplo de la figura 1.a.

Datos de salida

Para la salida de cada caso debe indicarse la cantidad mínima de movidas que se encontraron para lograr que se alineen las fichas y mostrar el tablero alineado con las fichas en la columna que soluciona el problema tal y como se muestra anteriormente en la figura 1.b.

Primer Proyecto Programado -Paradigma Imperativo, lenguaje C

Por ejemplo, para el primer caso del archivo procesado, la salida se mostraría:

Datos de entrada:

5

5 1 1 2 2 3 3 4 4 4

	1	2	3	4	5
1		●			
2			●		
3				●	
4				●	
5	●				

Datos de salida:

Tablero 1: 5 movidas requeridas

	1	2	3	4	5
1		➡	●		
2			●		
3			●	←	
4			●	←	
5	➡	➡	●		

Y así para cada uno de los casos leídos en el archivo.

Nota para los otros dos casos expuestos, las movidas necesarias serían:

Tablero 2: 0 movidas requeridas

Tablero 3: 1 movida requerida

Aspectos técnicos.

1. Deben utilizar el lenguaje de programación imperativo C. Pueden utilizar el IDE que deseen pero deben comunicarlo para efectos de revisión y aportar el ambiente requerido para dicho efecto.
2. Deben plantear sus soluciones de manera modular, es decir a través del uso de funciones que tengan una responsabilidad particular y que reciban por parámetro los insumos requeridos para llevar a cabo sus tareas.
3. No se permitirá el uso de variables globales, así que deberán hacer uso de los distintos tipos de paso de parámetros para el desarrollo de la tarea programada.

Primer Proyecto Programado -Paradigma Imperativo, lenguaje C

4. Puede si lo desea el equipo de trabajo, modularizar en distintos archivos las funcionalidades e incluirlas como parte de las invocaciones en el programa principal a través de una directiva `#include...` o bien establecer todas las funcionalidades en un único archivo.
5. Debe aportar la documentación interna de cada una de las funciones que se construyan para dar solución al problema planteado.
6. La ejecución del programa se inicia desde la línea de comandos.
Ejemplo: **alineafichasPruebasGrupoN.txt**

Entregables

1. Debe aportarse un archivo comprimido con extensión .zip en el que se incorpore:
 - a. Proyecto fuente que da solución al problema expuesto.
 - b. Archivo de pruebas utilizado para la validación y verificación del software construido.
 - c. Documentación adicional: informe escrito digital en el que presente los siguientes apartados:
 - i. Resumen del problema a resolver
 - ii. Estrategias y decisiones de diseño de la solución propuesta: ¿cómo decide el grupo resolver este problema?, ¿localización y uso de algoritmos particulares?, ¿modularización de la solución?
 - iii. Evidencia del uso de paso de parámetros por valor y por punteros.
 - iv. Evidencias de las pruebas realizadas.
 - v. Resumen de logros: enumeración de lo solicitado (100%), de este desglose cuánto se logra en forma completa, cuánto se queda pendiente y razones que justifiquen éstos últimos.
 - vi. Bibliografía o referencias digitales consultadas.
2. Archivo **readme.txt** con el detalle de los nombres y carnés de los tres estudiantes que integran el grupo, ambiente de desarrollo en el que fue desarrollado el proyecto, y ejemplo de la línea de comandos con la que debe invocarse el programa.

Aspectos Administrativos

1. Este proyecto puede ser realizado en grupos de 3 personas máximo.
2. Se entrega en el TEC Digital, los entregables solicitados en la sección de Evaluaciones>Proyectos>Proyecto 01 Imperativo el viernes 23 de marzo al final del día.
3. Cualquier intento de fraude en la construcción de este proyecto implica que los grupos involucrados obtienen una nota de cero en esta evaluación.
4. Se penalizará con 10 puntos si no se acatan las reglas de entrega tanto en formatos como en tiempos establecidos.
5. Las revisiones de este proyecto se agendarán la semana inmediata de la Semana Santa según solicitud previa que se registrará en un documento compartido que se publicará posteriormente.