

# Laboratorio 2

## Diseño de algoritmos

### Instrucciones generales:

- Debe ser realizado en forma individual.
- Se entrega en el aula virtual a más tardar el 20 de mayo de 2015. Se recuerda que no se revisarán entregas atrasadas.
- Se podrá realizar control de salida una vez entregado el laboratorio.
- El formato de entrega es un archivo comprimido con todos los códigos fuentes (proyecto si tiene) e informe en pdf (no se aceptará otro formato).
- Tendrá que hacer su trabajo el lenguajes de programación C++, incluyendo un archivo Makefile.
- Si realiza su informe en  $\text{\LaTeX}$ , tendrá puntaje extra.
- Debe utilizar el método de resolución de fuerza bruta.

### Objetivos

- Crear, diseñar e implementar un programa en fuerza bruta.

### Enunciado

El juego Flow <sup>1</sup> o el juego Connect The Dots <sup>2</sup> se tratan de unir 2 puntos del tablero del mismo color, en donde los movimientos posibles son vertical u horizontal. Al unir los puntos se trazan líneas en donde ya no es posible que otra línea de otro color pase por ahí. Para esta versión no va a ser necesario que el tablero entero quede totalmente lleno de líneas.

Su misión será dar solución a estos juegos por medio de un programa.

### Entrada

La entrada corresponde a múltiples casos de pruebas. En la primera línea van dos números, el primero es el tamaño del tablero cuadrado  $N$  ( $3 \leq N \leq 100$ ) y el siguiente corresponde a la cantidad de colores  $C$  ( $2 \leq C \leq 500$ ) que se van a utilizar para el tablero, en donde  $C \leq N^2$ ). Las siguientes  $N$  líneas corresponde a cada fila del tablero. En cada columna va a ir un número correspondiente al color o una línea — para indicar que no hay un punto en esa posición. Cada número va a aparecer exactamente dos veces en el tablero. Los números parten desde el 1. Si  $N = 0$  se terminan los casos de prueba. Todos los números están separados por un espacio. Entre cada caso de prueba está separado por un enter.

---

<sup>1</sup><https://play.google.com/store/apps/details?id=com.bigduckgames.flow>

<sup>2</sup><https://play.google.com/store/apps/details?id=pub.connectthedots>

## Salida

En la primera línea de cada caso de prueba, deberá colocar: “CASO # $X$ ”, donde  $X$  corresponde al caso de prueba, partiendo desde 1. A partir de la siguiente línea, deberá dibujar el tablero y en las celdas que unen colores (las que inicialmente están con  $-$ ), colocar el número al que están uniendo.

En caso de que no encuentre solución, deberá colocar el mensaje “NO TIENE SOLUCIÓN”. Puede darse el caso que existan más de una solución. Para ello, entregar una solución al problema. Entre casos separarlos con un enter.

En la Tabla 1 se muestra un ejemplo de entrada y salida del programa.

## Informe

Debe ser escrito en tercera persona y tiempo presente. El formato de impresión a utilizar es el de presentación de memoria de título, encontrado en el aula virtual.

Debe contener los siguientes elementos:

- Portada.
- Índice.
- Introducción (1 página): establece la motivación y los objetivos del trabajo. Indicar en qué máquina (características) se realizaron las pruebas. Además describe brevemente la estructura del resto del documento.
- Marco teórico.
- Desarrollo del trabajo:
  - Algoritmos en pseudocódigo y breve explicación.
  - Detalle de las instancias utilizadas o cómo crearon las instancias para las pruebas (usar columnas para disminuir la cantidad de páginas).
- Conclusión: presenta un análisis de los resultados obtenidos e indica si se cumplieron los objetivos planteados en la introducción.
- Referencias: deben ser escritas según estándar APA y deben aparecer debidamente señaladas en el documento.

## Requerimientos

1. Debe entregar su código, se evaluará el orden, la separación en funciones, la mnemotecnía y el uso de comentarios.
2. Los documentos deberán ser presentados impresos en papel y en formato pdf.

## Observaciones importantes

El informe que debe entregar, tiene que mostrar claramente las razones de las decisiones que usted tomo en el sistema, tanto de navegación, interfaz de usuario, funcionalidades que usted presenta. Además el informe debe mostrar las decisiones tomadas con respecto a la implementación y uso del lenguaje.

El sistema debe ser robusto, **se penalizarán las caídas** de cualquier tipo. Debe además estar bien documentado, si así lo desea puede presentar la documentación creada con Doxygen (no es obligatorio).

**La copia** de cualquier tipo (incluso de internet) **será evaluada con un 1.0**. Se han dejado a propósito algunos temas a su criterio, la intención es que usted sea autocrítico en su trabajo y se esfuerce en entregar el mejor sistema (incluidos documentos) posible, por lo que un sistema que sólo cumpla con los criterios mínimos mencionados en este enunciado no aseguran un 7, recuerde, sea eficiente, no sólo eficaz.

El enunciado podría sufrir modificaciones.

Tabla 1: Ejemplo

Entrada	Salida
5 5 1 2 3 4 5 - - - - - - - - - - - - - - - 1 2 3 4 5	CASO #1 1 2 3 4 5 1 2 3 4 5 1 2 3 4 5 1 2 3 4 5 1 2 3 4 5
3 3 1 3 2 - - - 2 3 1	CASO #2 NO TIENE SOLUCIÓN
5 5 1 2 - - - - - - - 2 3 - 4 - 1 5 3 - 4 5 - - - - -	CASO #3 1 2 2 2 2 1 1 1 1 2 3 3 4 1 1 5 3 4 4 5 5 5 5 5 5
8 9 1 - - 1 2 - - - - - 3 4 5 - 5 - - - 6 - - - - - - - 7 - 8 9 - - - - - - 4 - - - - - - 3 - - - - - 6 - 7 - - 8 2 - - - - - - 9	CASO #4 1 1 1 1 2 2 2 2 3 3 3 4 5 5 5 2 3 6 6 4 8 8 8 2 3 6 7 4 8 9 8 2 3 6 7 4 4 9 8 2 3 6 7 3 3 9 8 2 3 6 7 7 3 9 8 2 3 3 3 3 3 9 9 9
6 3 - - - - - - 3 - - 2 - - 1 - - - - 2 1 - - - 3 - - - - - - - - - - - - - 0	CASO #5 - - - - - - 3 3 3 2 2 2 1 1 3 3 3 2 1 1 - - 3 - - - - - - - - - - - - -