



# Camino más Corto

## Laboratorio Semana 7

### 1 Introducción

La cadena de restaurantes temáticos *Pizza Planeta*, debido a su elaborada decoración, tiene limitados los caminos entre las mesas. Sin embargo, la gerencia cree que sus meseros están llevando las pizzas a la mesa usando precisamente el camino más largo. Usted ha sido elegido para encontrar el camino más corto a cada mesa, dado un grafo y la ubicación de la cocina. Los meseros pueden ir y regresar por los caminos.

### 2 Descripción de la Actividad

Se desea encontrar el camino de costo mínimo desde un vértice inicial hasta todos los vértices de un grafo no dirigido.

Debe crear un archivo llamado `Mesero.java` con la implementación de la aplicación.

### 3 Entrada de datos

El programa se debe poder ejecutar desde la consola con el siguiente comando:

```
>java Mesero <instancia> <origen>
```

donde `<instancia>` es el nombre del archivo que contiene el grafo, y `<origen>` es el número del vértice que corresponde a la cocina,

El formato del archivo que contiene los datos de un grafo es el siguiente:

```
n                //número de nodos
x0 y0          //posición del nodo 0
:
xn-1 yn-1      //posición del nodo n-1

m                //número de lados
nodoIni0 nodoFin0 //lado 0 (nodoIni nodoFin)
:
nodoInim-1 nodoFinm-1 //lado m-1 (nodoIni nodoFin)
```

Para el costo de los lados, utilice la distancia euclidiana.

### 4 Salida de los Datos

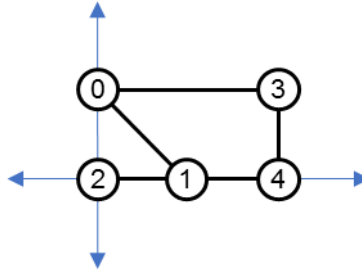
La salida de su programa deberá mostrar, para cada vértice, el camino para llegar a él, su longitud, y su costo en el siguiente formato:

```
Nodo <i>: n1->n2->...->nL    <L> lados (costo <C>)
```

Puede usar el caracter especial ‘\t’ para separar el camino del número de lados

## 5 Ejemplo

Considere el siguiente grafo no dirigido, el cual se proporciona en el archivo `ejemploA.txt`:



El resultado de la llamada `Mesero ejemploA.txt 2` debería ser:

```
Nodo 0: 2->1->0      2 lados (costo 2.41)
Nodo 1: 2->1          1 lados (costo 1.0)
Nodo 2: 2             0 lados (costo 0.0)
Nodo 3: 2->1->4->3    3 lados (costo 3.0)
Nodo 4: 2->1->4        2 lados (costo 2.0)
```

Puede usar la función `format` de `DecimalFormat` (creada con `new DecimalFormat("0.0#")` en este ejemplo) para fijar el número de decimales de su impresión.

## 6 Entrega

Debe entregar, a través del Moodle antes del lunes, 12 de noviembre a las 11:55 pm. un archivo comprimido que debe contener los archivos fuentes del código de su solución.

## 7 Evaluación

De ser escogido para evaluación, se asignarán:

- 5 puntos por código
  - 2 puntos por su algoritmo de camino de costo mínimo
  - 1 punto por utilizar coordenadas cartesianas correctamente
  - 1 punto por utilizar distancia euclídeana correctamente
  - 1 punto por su lectura de archivo
- 4 puntos por ejecución
  - 1 punto por imprimir la ruta desde la cocina hasta cada mesa en el formato correcto
  - 1 punto por imprimir la ruta más corta
  - 1 punto por imprimir la longitud correcta del camino
  - 1 punto por imprimir el costo correctamente
- 1 punto por documentación

Si usa `DecimalFormat` correctamente, se asignará un punto adicional.