Proyecto 1: Rutas Óptimas (Algoritmo de Floyd)

Jueves 23 de Agosto

I. Descripción General

Se debe programar el Algoritmo de Floyd para obtener las rutas óptimas entre cualquier par de nodos en un grafo ponderado con distancias. La interfaz debe ser gráfica. Toda la programación debe realizarse en C sobre Linux. No se pueden cambiar las especificaciones de este documento.

II. Entrada y Salida

La interacción con el usuario se hará por medio de interfaces gráficas que deben ser desarrolladas con `GTK` y `Glade`. Debe haber un estilo uniforme en el *look & feel* de todas las interfaces. El programa tendrá la posibilidad de grabar archivos con los datos particulares de cada problema ingresado por el usuario, para que puedan ser cargados de nuevo y editados si el usuario así lo desea. El formato de este archivo queda a discreción de cada grupo de trabajo. Se espera que, el día de la revisión, cada grupo cuente con bastantes archivos de pruebas para mostrar las capacidades de su proyecto.

III. Menú Principal

En el Proyecto 0 de este curso se desarrolló un menú principal que podía "lanzar" diversos algoritmos. El algoritmo solicitado en este proyecto será implementado como un programa independiente que será ejecutado desde el menú mencionado. Se debe activar una "burbujita" con una descripción general de este algoritmo cuando el cursor se pose sobre la opción respectiva en el menú principal (*tooltip*).

IV. Problema de las rutas más cortas

Usando el algoritmo de Floyd estudiado en clases, este programa resolverá el problema de encontrar las rutas más cortas entre cualquier par de nodos de un grafo con ponderaciones en los arcos.

Se espera que la interfaz gráfica sea lo más flexible posible. El usuario debe proporcionar la cantidad de nodos del grafo y las distancias directas entre ellos. Debe haber un mecanismo para indicar el caso en que las distancias son infinitas (*i.e.*, no ruta directa) entre 2 nodos. Este es el valor por defecto para todas las entradas de la tabla al inicio (excepto la diagonal de la tabla que tendrá siempre ceros)[1]. Para este proyecto, el usuario podrá indicar entre 1 y 10 nodos.

Trabajo extra opcional 1: hacer que se manejen apropiadamente más de 10 nodos.

Cada nodo tendrá un nombre editable pero por defecto se identificarán como A, B, C, etc.

[1]No usen -1 como infinito, sean creativos

Conforme se ingresan los datos de distancias entre nodos, el programa despliega en la interfaz gráfica el grafo en su estado actual. Se sugiere revisar "Cairo".

Esencialmente, el usuario ingresará los datos de $D(0)$. A continuación, usando algún mecanismo apropiado (*e.g.*, un botón) el usuario solicitará el cálculo de las siguientes tablas intermedias ($D(1)$, $D(2)$, $D(3)$, etc.) hasta llegar a la tabla final. Cada tabla mostrará de una manera apropiada los cambios con respecto a la tabla anterior. También en cada iteración se mostrará el estado actual de la tabla $P(k)$ de rutas.

Después de la última tabla, se activará un diálogo que permite solicitar la ruta óptima entre cualquier par de nodos. Esta información está codificada en la tabla P.

V. Requisitos Indispensables

La ausencia de uno solo de los siguientes requisitos vuelve al proyecto "no revisable" y recibe un 0 de calificación inmediata:

- Todo el código debe estar escrito en C
- El proyecto debe compilar y ejecutar en Linux
- Todas las interfaces deben ser gráficas
- Se debe usar `GTK` y `Glade`
- El programa se debe invocar desde el menú desarrollado en el Proyecto 0.

VI. Fecha de Entrega

Revisiones a las 11:30am del **Jueves 23 de Agosto**. Mande además un `.tgz` con todo lo necesario (fuentes, makefile, readme, etc.) a `torresrojas.cursos@gmail.com`. Ponga como subject: `I.O. - Proyecto 1 - Fulano - Mengano`, donde Fulano y Mengano son los 2 miembros del grupo.

Mucha suerte...