

EIF207 – Estructuras de Datos

Proyecto de programación #2

Prof. Georges E. Alfaro Salazar
Prof. José Pablo Calvo

OBJETIVOS DEL PROYECTO

El objetivo del proyecto es aplicar los conceptos teóricos, principios y técnicas estudiadas en clase. Es importante que al desarrollar el proyecto se escriba código correctamente estructurado y encapsulado. El proyecto busca en particular que los estudiantes implementen **estructuras de datos no elementales** y que realicen un análisis adecuado de la eficiencia de los algoritmos utilizados (tanto en tiempo como en espacio).

DESCRIPCIÓN DEL PROYECTO

Para este proyecto, se solicita **realizar una aplicación que calcule rutas y permita simular una operación de reparto**. El sistema está siendo desarrollado por una compañía que desea **implementar un modelo de verificación de rutas más cortas que sustituya el uso de Waze** para los repartidores de varias empresas. El sistema permitirá agilizar la forma en que los repartidores hacen el recorrido hasta su destino.

FUNCIONALIDADES POR IMPLEMENTAR

Se modelará el mapa de una ciudad que cuente con carreteras, localizaciones y controles de ruta (señales y semáforos). Se **simulará el recorrido de uno o varios repartidores** para mostrar la idoneidad de las rutas planeadas.

El programa **mostrará un mapa simple**, que **se cargará desde algún archivo con la información correspondiente**.



Las diferentes **localizaciones en el mapa tendrán un nombre descriptivo**. Use imágenes (iconos) para identificar elementos especiales, como semáforos y señales.

Cada una de las localizaciones representadas, corresponden a un nodo del grafo (vértice). También existe un vértice en cada punto de intersección que sea necesario para conectar las rutas en el mapa. El mapa mostrará las rutas entre cada localización. **Cada ruta tendrá un peso que puede asignarse de manera**

aleatoria o utilizando un valor por defecto apropiado. Hay que definir un rango para los valores asociados al peso de cada ruta (arista o arco). El peso de cada arco representará el costo de transitar esa ruta en particular. Puede estar asociado con el tiempo necesario para recorrer la ruta o su distancia, así como la cantidad de “tráfico”, correspondiente a la cantidad de elementos que se encuentran en dicha ruta.

Los nodos (vértices) podrán contar con semáforos, que limitan el paso de una ruta a otra.

El programa permitirá indicar o ingresar puntos de partida y destino para un repartidor. Una vez que la información es ingresada, se mostrará en tiempo real (simulado usando un factor de escala de tiempo). En un momento cualquiera, la aplicación mostrará cada uno de los repartidores moviéndose sobre las diferentes rutas. Observe que el peso asociado a cada ruta puede variar de acuerdo con la cantidad de elementos trasladándose sobre esa ruta.

El propósito del programa es calcular el costo de las rutas más cortas entre cada punto de partida y destino. Las rutas se mostrarán de manera tabular, indicando los costos respectivos. Cuando un nuevo repartidor es ingresado para recorrer una ruta que ya ha sido transitada anteriormente, éste seleccionará dicha ruta de manera preferente, e irá realzando los ajustes necesarios según la información de tráfico.

CONSIDERACIONES DE IMPLEMENTACIÓN

Para este simulador se usará un grafo conexo ponderado y dirigido para representar el mapa por donde circularán los diferentes elementos (repartidores).

Las rutas se mostrarán sobre el mapa (una vez que se hayan recorrido en su totalidad) usando una flecha, donde la punta indicará el sentido de la ruta.



Para implementar el grafo, puede utilizar una lista de adyacencia o listas de arcos, como los considere más conveniente. Recuerde que la aplicación NO permite editar el grafo, que se recupera desde un archivo. Es recomendable usar un archivo XML para la información del grafo, ya que permite ser editado fácilmente. Deberá incluir al menos tres (3) mapas de prueba para la aplicación.

Puede agregar elementos gráficos que mejoren el aspecto estético del grafo, tales como ríos, árboles, esquemas de edificios o elementos similares.

La simulación debe mostrar a los repartidores moviéndose a través del mapa de una localización a otra empleando la ruta más corta. Utilice alguna variación del algoritmo de Dijkstra para recorrer el grafo y obtener la ruta más corta de un punto a otro. Observe que cómo las características del grafo cambian, es posible incluso que un repartidor deba retroceder en su ruta o cambiar una ruta planeada.

El programa deberá:

- Leer los datos del grafo desde un archivo seleccionable por el usuario.
- Incluir repartidores y designar rutas.
- Iniciar o detener la simulación.
- Mostrar de manera tabular los datos de las rutas utilizadas.

REFERENCIAS.

Puede consultar las siguientes referencias:

- <https://sites.google.com/site/maticasmoralesgalindo/6-1-elementos-y-caracteristicas-de-los-grafos/6-1-2-tipos-de-grafos-simples-completos-bipartidos-planos-conexos-ponderados>
- <http://www.oia.unsam.edu.ar/wp-content/uploads/2017/11/dijkstra-prim.pdf>

ENTREGA Y EVALUACIÓN

El proyecto debe entregarse **por medio del aula virtual, en el espacio asignado para ello, o de la manera solicitada por el profesor del curso**. La fecha de entrega es el día **viernes 16 de noviembre de 2019**. No se aceptará ningún proyecto después de esa fecha, ni se admitirá la entrega del proyecto por correo electrónico. **El proyecto puede completarse en grupos de 2 a 3 personas. Alguno de los integrantes de grupo entregará un archivo de texto con el detalle de los participantes, a más tardar el día 2 de noviembre de 2019. No se permitirá que ninguna persona se integre posteriormente a cualquier grupo de trabajo.**

Incluya comentarios en el código de los programas y describa cada una de las clases y métodos utilizados cuando sea útil y conveniente.

El código del programa debe compilar correctamente. Si el código no puede compilarse, el trabajo será calificado con una nota de 0 (cero). Para ser evaluado, tiene que cumplirse al menos un 50% de la funcionalidad solicitada. De lo contrario, el proyecto será considerado como no entregado y se penalizará la nota correspondiente.

Durante la revisión del proyecto, es muy importante defender adecuadamente la solución propuesta.

Se evaluarán los siguientes aspectos:

Documentación	5%
Aspectos formales (presentación general, estructura, redacción y ortografía)	1%
Descripción de la solución	2%
Correctitud y completitud de los diagramas	2%
Estructura	35%
Carga de los datos del mapa	10%
Definición de clases e implementación de estructuras	25%
Funcionalidad	60%
Dibujo del grafo conexo y dirigido	15%
Ingreso de datos (repartidores, puntos de inicio y llegada)	10%
Funcionamiento de la simulación (animación y control)	20%
Cálculo dinámico del camino más corto y caminos alternos	15%

Observaciones generales:

- Los proyectos deben entregarse con toda la documentación, diagramas, código fuente y cualquier otro material solicitado.
- Los trabajos no se copiarán de ninguna llave USB u otro dispositivo en el momento, sino que se deben entregar en el formato adecuado.
- **Cualquier trabajo práctico, que no sea de elaboración original del estudiante y haya sido copiado o adaptado de otro origen (plagio), de manera parcial o total, se calificará con nota 0 (cero) y se procederá como lo indiquen los reglamentos vigentes de la universidad.**