Segundo Proyecto Programado

Estructura de Datos

Profesora Lorena Valerio.

I Semestre 2016

**Aplicación para el uso de la Estructura Grafo, árbol**

**y archivos**

**Objetivos**

Desarrollar un programa en el lenguaje c++, utilizado las ventajas de OO.

Implementar las estructura de datos tipo red: Grafos.

Implementar estructuras indexadas para búsquedas más eficientes.

**Descripción**

Implementar una estructura de datos grafo no dirigido para una aplicación de transporte, además debe administrar clientes y envío de encomiendas. Se deberan repartir las encomiendas asignadas escogiendo la ruta más corta y dejandolas en el vertice que corresponda. El programa deberá calcular lo que debe pagar el cliente, según peso y el kilometraje de la ruta hasta el vertice destino de la encomienda.

La información de los clientes y encomiendas deberán ser almacenada en archivos. Al iniciar el programa debe cargar (desde el código) los datos del grafo de transporte, crear el árbol binario ordenado por id de la lectura del archivo de encomiendas. Ademas debe solicitar los paquetes de máximo 5 encomiendas diferentes que deben entregarse y generar la facturación por cada cliente. El programa debe cuidar que el cliente pague lo menos posible por las entregas de las encomiendas (ruta corta), se podrá indicar de donde inicia el recorrido de la ruta. Si los archivos estan vacíos el programa debe permitir el ingreso de los datos respectivos a los archivos correspondientes.

**Tipo de datos abstracto del grafo**

*Vértice*: representa las diferentes ciudades, contiene información como nombre de la ciudad.

*Arco:*  representa los caminos u autopistas entre las ciudades, cada arco contiene información sobre el destino y la distancia en km.

**Tipo de datos abstracto árbol binario**

Mapa

*Encomiendas:* representan los envíos que se deben traladar atrás del grafo de transporte*,*  se guarda información como: id, peso, descripción, un apuntador hacia que cliente destinatario, la ciudad destino y sus enlaces izquierda y derecha.

**Tipo de datos abstracto lista o arreglo**

Paquete de encomiendas: registrara las encomiendas que deben ser entregadas, en cada elemento de la lista se guardará el nodo del arbol binario y un puntero sig.

**Archivos**

1. **Archivo clientes,** el acceso a este archivo es secuencial, sus atributos son: cedula, nombre, dirección y un puntero a su primera encomienda.
2. **Archivo encomiendas,** el acceso este archivo es directo, desde el archivo de clientes. El cual será cargado en un arbol binario en memoria 3principal. Sus atributos son: id, peso, descripción, ciudad destino y siguiente encomienda del mismo cliente.

**Ejemplos de los archivos:**

**Archivo Clientes**

|  |  |  |  |
| --- | --- | --- | --- |
| ced | nombre del cliente | Dirección | enlace archivo  de encomiendas |
| 30 | Andres | Alajuela | 0 |
| 20 | Jurguen | Heredia | 1 |
| 40 | Carlos | San\_Carlos | 2 |
| 25 | Esteban | San\_Jose | 3 |
| 38 | Roberto | Limon | 4 |

**Archivo Encomiendas**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **reg** | **id** | **peso** | **descripción** | **Destino** | **sig** |
| 0 | 10 | 10 libras | Ropa | San\_Jose | 5 |
| 1 | 12 | 1 gramos | Tenis | Alajuela | 4 |
| 2 | 1 | 2 k | libros | Heredia | 6 |
| 3 | 56 | 10 12k | Vajilla | Cartago | -1 |
| 4 | 3 | 5 | Ropa de cama | Limon | -1 |
| 5 | 4 | 1 | Libro | Guanacaste | -1 |
| 6 | 8 | 1 | Perfumes | Puntarenas | -1 |

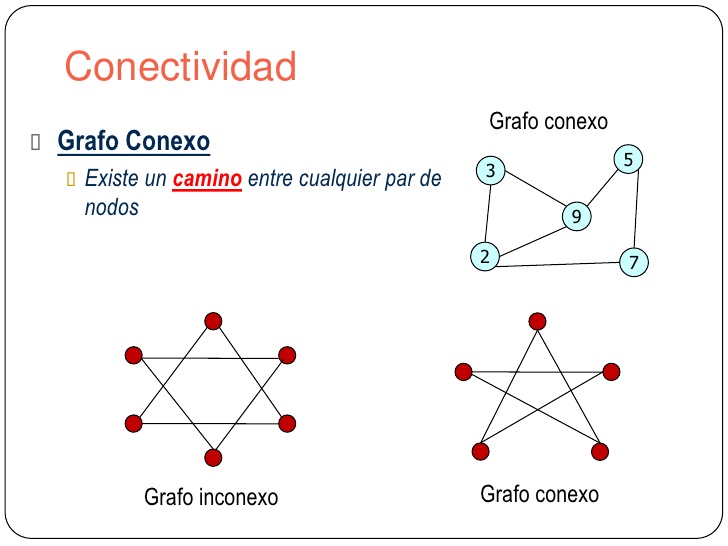
**Operaciones TDA**

Debe realizar operaciones sobre los grafos como:

1. Insertar y borrar ciudades, en cualquier momento del programa.
2. Insertar, modificar y borrar caminos.

**Operaciones sobre los archivos**

1. Insertar clientes.
2. Insertar encomiendas.

**Consultas**

1. Determinar si el grafo de transporte es conexo: si existe un arco entre cualquier par de vértices.
2. Imprimir todas las encomiendas de un cliente X. Debe acceder al archivo de forma eficiente usando los respectivos punteros.
3. Imprimir la ruta corta (utilizando la distancia) escogida para entregar el paquete completo de encomiendas, con su respectiva distancia total, indicar el orden de entrega de cada una de las encomiendas del paquete, que mejor se ajusto para obetner la ruta corta.

Ejemplo: Ruta Corta es: A, CQ, D, F, G, SR= 35 km

1. La encomienda id= 10 se dejo en CQ.
2. La encomienda id= 56 se dejo en F.
3. La encomienda id= 3 se dejo en SR.

**Nota**: para esta consulta debe pedirle al usuario que haga una paquete indicando los id de las encomiendas (máximo 5) que va enviar, además debe pedirle de que vertice va empezar el recorrido.

**Reportes**

1. Imprimir la factura para cada cliente una vez realizada la entrega de las encomiendas. En la misma factura se incluiran todas las encomiendas entregadas que pertencen a un mismo cliente. La impresión en consola debe tener formato de factura. El costo será por peso de la encomienda y kilometraje recorrido.
2. Imprimir el Archivo de clientes, tal y como se ve en la tablas anteriores.
3. Imprimir el Archivo de encomiendas, tal y como se ve en las tablas anteriores.
4. Imprimir el arbol binario con toda la información en “preOrden” y “enOrden”.

**Interfaz de usuario**

En consola, realizar un menú, con las diferentes opciones para acceder a cada una de las funciones del programa.

**Documentación Externa**

Debe seguir la guía de Estandarización para las Documentaciones Solicitadas en Proyectos Programados de la Unidad Desconcentrada de Computación.

**Documentación Interna**

* Fecha de inicio y Fecha última modificación, solo en el archivo del main principal.
* Descripción para cada clase o estructura y su uso en el programa.
* Describir cada función. Indicar que hace, que recibe, y que devuelve.
* La Notación a usar es CamelCase: LowerCamelCase.

**Aspectos Administrativos**

* La tarea debe programarse en lenguaje C++.
* Debe traer datos cargados en el código del programa fuente y permitir la carga por medio de archivos.
  + El desarrollo de este trabajo se puede realizar en pareja como máximo.
  + Entrega de la tarea: martes 9 de junio a las 23:00.
  + Se calificará con citas de revisión para la defensa de la tarea.
  + Si se encuentra copia la calificación será de cero para todos los implicados.
  + Todo proyecto y documentación debe entregarse el link respectivo del tecDigital.
  + Si no abre el proyecto no se calificara la parte programada.
  + Se recomienda que se comience a trabajar desde hoy.