

Algoritmos

Trabajo Práctico

El trabajo práctico consiste en realizar una aplicación para el abastecimiento de estaciones de servicio.

La información de las estaciones estará dada en un archivo de texto estaciones.txt que deberá leerse al iniciar la aplicación.

En cada línea del archivo estará la información de cada estacion, separada por espacios:

```
codigo nombre ciudad #cantidad_surtidores #litros_surtidor#tipo_combustible
```

El código son 3 letras y 3 numeros que identifica a cada estacion. Por ejemplo, BAS001 es la estacion 1 de Buenos Aires, COR003 es la estacion 3 de Córdoba.

La cantidad de surtidores indica cuantos surtidores de un determinado tipo de combustible (Cada surtidor tiene una capacidad maxima de 100 litros) tiene la estacion. Los tipos de combustibles son SUP (Super), INF(Infinia), NIT(Nitro).

Ejemplo:

```
BAS001 Estacion1 Bs As 4 SUP
BAS001 Estacion1 Bs As 3 INF
BAS001 Estacion1 Bs As 2 NIT
COR003 Estacion3 Cordoba 6 SUP
COR003 Estacion3 Cordoba 4 INF
COR003 Estacion3 Cordoba 4 NIT
```

Se debe mostrar un menú en el cual se pueda:

- consultar la informacion de una estacion en particular.
- dar de alta una nueva estacion.
- dar de baja estaciones.
- mostrar todas las estaciones (ordenados por los distintos datos) y totalizando por distintos tipos de datos.

Consideraciones

- El archivo está bien formado.
- La implementación debe utilizar una tabla de hashing.
- La función de hashing a utilizar debe lograr una buena dispersión.

Viajes

A partir de la lectura de un archivo de texto viajes.txt, que deberá hacerse inmediatamente después de leer el archivo estaciones.txt, se generará un grafo pesado dirigido. Este archivo contiene los viajes que puede hacer un camion cisterna.

En cada línea estará la información de cada viaje de estacion a estacion, separada por espacios:

```
codigo_partida código_destino costo_viaje horas_viaje
```

El costo del viaje es un entero que representa el costo en pesos y las horas de viaje será un flotante.

Ejemplo:

```
BAS001 COR003 22483 7.31
COR003 USU002 222205 46
BAS001 ROS005 14736 3.25
...
```

La aplicación debe pedir al usuario el ingreso de los códigos de partida y de destino, y debe buscar la combinación de viajes más económica o de menor tiempo. El usuario debe poder elegir si el camino mínimo está representado por el costo del viaje o por la duración. Se debe imprimir el detalle del viaje, el costo de cada tramo (o el tiempo de viaje) y el total

del viaje. Si hubiera más de una combinación óptima, deberá listarlas todas. Puede suceder que no haya ninguna combinación, en ese caso se informa que no hay conexiones posibles.

Es importante tener en cuenta que los camiones cisterna tienen una capacidad limitada de 25000 litros para los 5 tipos de combustible.

También es importante considerar que en cada viaje el camión no debe quedarse sin combustible para abastecer las estaciones de servicio que visita.

Consideraciones

- El archivo está bien formado.

Tener en cuenta

- Funcionalidad
- Interfaz con el usuario
- Nombres de métodos y atributos
- Modularización
- Buenas prácticas de programación
- Diseño UML
- Complejidad algorítmica

Nota: la aplicación debe estar completamente orientada a objetos, y el lenguaje utilizado debe ser C++. Debe hacerse correcto uso de la memoria dinámica.

Entrega

Semanalmente cada equipo deberá realizar una exposición del avance del proyecto.

La entrega final del trabajo será en horario de clases, momento en el que se hará la defensa de este. Por defensa se entiende la muestra del trabajo junto con una explicación por parte de cada uno de los integrantes.