

75.29 Teoría de Algoritmos I

Trabajo Práctico Nº 2

Fecha de Entrega: 14/11/2014

Viaje en Tren (programación dinámica):

María que vive en la ciudad i quiere participar del concurso de programación que se llevará a cabo en la ciudad j .

Mañana se inscribirán a las primeras personas que se presenten en j , por lo que María debe preparar su viaje para llegar lo antes posible.

Sin embargo a María no le gusta esperar mucho en las estaciones, así que si hay varios trenes que lleguen a la misma hora, María elegirá el que salga más tarde.

María tiene los horarios de los trenes y les pide a Uds. que la ayuden a elegir las combinaciones tal de llegar lo antes posible a j y con las conexiones más rápida posibles. Como tiene poco equipaje no demora nada en cada trasbordo.

Archivo de entrada:

Es un archivo de texto con el siguiente formato:

- La primera línea describe la cantidad de escenarios. Cada escenario cuenta con tres partes:
 - Una línea con la cantidad de ciudades N ($1 \leq N \leq 100$). Seguida de N líneas, con el nombre de las ciudades.
 - La segunda parte del escenario describe todos los trenes del día. Comienza con un entero $T \leq 1000$ que indica la cantidad de trenes programado para ese día, seguida de T descripciones de horarios. Cada una de las cuales consiste en:
 - Una línea con un entero $t_i \leq 100$ seguida de t_i líneas más
 - Cada una de estas t_i líneas contiene una hora y un nombre de ciudad, indicando que el tren t_i para en esa ciudad a esa hora.
 - Por último la tercera parte consiste en tres líneas:
 - La primera la hora a partir de la cual puede María puede iniciar la travesía
 - La segunda línea es la ciudad i (ciudad de origen de María)
 - La tercera línea es la ciudad j donde se lleva a cabo el concurso (ciudad destino)

Salida:

Para cada escenario imprimir una línea con el número de escenario comenzando en 1.

Si hay conexión imprimir dos líneas con las horas y ubicaciones.

Si no hay conexión en el mismo día imprimir una línea que indica sin combinaciones posibles.

Dejar una línea en blanco entre dos escenarios

Ejemplo:

2	←Escenarios
3	←Cantidad de ciudades del escenario 1
Jujuy	←Nombre de ciudad
Tucumán	←Nombre de ciudad
Buenos Aires	←Nombre de ciudad
4	←Cantidad de trenes programados en el día de la fecha
2	←Cantidad de ciudades que visita el tren 1
0949 Jujuy	←Hora y ciudad de partida del tren 1
1006 Tucumán	←Hora y ciudad de llegada del tren 1
3	←Cantidad de ciudades que visita el tren 2
1000 Jujuy	←Hora y ciudad de partida del tren 2
12:00 Tucumán	←Hora y ciudad con parada intermedia del tren 2
1550 Buenos Aires	←Hora y ciudad de llegada del tren 2
2	←Cantidad de ciudades que visita el tren 3
1205 Tucumán	←Hora y ciudad de partida del tren 3
1411 Buenos Aires	←Hora y ciudad de llegada del tren 3
2	←Cantidad de ciudades que visita el tren 4
0700 Jujuy	←Hora y ciudad de partida del tren 4
12:00 Buenos Aires	←Hora y ciudad de llegada del tren 4
0800	←Hora a partir de la cual María puede comenzar la travesía
Jujuy	←Ciudad de origen de María
Buenos Aires	←Ciudad destino donde se realiza el concurso
2	←Cantidad de ciudades del escenario 2
Córdoba	←Nombre de ciudad
La Plata	←Nombre de ciudad
1	←Cantidad de trenes programados en el día de la fecha
2	←Cantidad de ciudades que visita el tren 1
0100 Córdoba	←Hora y ciudad de partida del tren 1
0600 La Plata	←Hora y ciudad de llegada del tren 1
0800	←Hora a partir de la cual María puede comenzar la travesía
Córdoba	←Ciudad de origen de María
La Plata	←Ciudad destino donde se realiza el concurso

Salida

Escenario 1
Salida 1000 Jujuy
Trasbordo 1200 Tucumán
Arribo 1411 Buenos Aires

Escenario 2
Sin combinaciones posibles

Informe:

Se debe presentar un informe impreso con el análisis, diseño y código fuente de todos los algoritmos desarrollados (incluyendo funciones o métodos auxiliares para leer archivos, etc.), se debe incluir diagramas de clase y cálculo de órdenes de todos los algoritmos implementados.

Se deberá enviar por mail a mfranzo+tda2014+tp2@gmail.com el informe anterior en formato .pdf, el código fuente con archivo para compilación automática (Makefile, build.xml, etc.) una versión compilada para linux, un readme con instrucciones de uso y varios archivos .txt con datos de prueba.

Todos los programas desarrollados se tienen que poder ejecutar en las máquinas de la facultad (con Linux)