

En la empresa de autobuses “Transportes y Viajes Rápidos Confort Interurbano de Destinos Variados y Escapadas Recreativas Sociedad Anónima de Responsabilidad Limitada Hermanos Machado” (T.Y.V.R.C.I.D.V.E.R.S.A.R.L.H.M.) se desea gestionar las líneas o trayectos que hacen sus autobuses. Para ello, se almacenarán las líneas que se hacen (una línea es un viaje que se hace diariamente, por ejemplo, el trayecto Madrid-Barcelona), y los kilómetros que tiene cada línea.

Se debe hacer un programa que muestre un menú y permita almacenar los datos, según se explica más adelante. El programa debe permitir al usuario la introducción para una línea (trayecto), de la ciudad origen la ciudad destino y los kilómetros. Hay que tener en cuenta que por cuestiones logísticas, puede haber una línea (trayecto) que haga el viaje de Sevilla a Cádiz, pero no tiene por qué haber siempre (puede haberla o no), la línea inversa, es decir, de Cádiz a Sevilla. Si la hay, se introducirá como una línea distinta, además, los kilómetros pueden ser distintos en un sentido que en otro (por cuestiones de tráfico, en un sentido se puede ir por una carretera y en el otro sentido por otra distinta).

Para ello, se mostrará el siguiente menú (*1 punto, junto con la valoración general del programa*):

- 1 - Añadir una línea de una ciudad a otra.**
- 2 - Listado de todos los datos introducidos.**
- 3 - Mostrar si hay ida y vuelta entre dos ciudades dadas.**
- 4 - Listar los trayectos más cortos que una distancia dada desde una ciudad dada.**
- 5 - Comprobar si se puede ir de una ciudad dada a otra dada en justo dos viajes.**
- 6 - Fin.**

Opción 1: (*2 puntos*) pedirá el nombre de la ciudad origen, de la ciudad de destino y los kilómetros del origen al destino. Se mostrará un mensaje de “datos almacenados” y se volverá al menú. En este apartado sólo se introduce una línea de autobús, es decir, origen, destino y kilómetros. No varios trayectos. No se van a manejar más de 30 ciudades distintas. El usuario no va a cometer errores; no va a introducir como origen y destino la misma ciudad, la distancia va a ser un número entero, mayor que cero.

Opción 2: (*1 puntos*) mostrará un listado de todos los datos almacenados, como se muestra en el ejemplo en la siguiente página.

Opción 3: (*2 puntos*) Pedirá al usuario el nombre de una ciudad origen A, de una ciudad destino B, y mostrará si hay un trayecto desde A hasta B y otro desde B a A. Podemos suponer que el usuario no va a cometer errores, y las ciudades A y B están previamente introducidas.

Opción 4: (*2 puntos*) Pedirá al usuario el nombre de una ciudad origen A y una distancia en kilómetros, K. Mostrará las ciudades hasta las que hay una línea (trayecto) que parta desde A y que tenga menos de K kilómetros. Si no hay ninguna, mostrará “ninguna”. Podemos suponer que el usuario no va a cometer errores, y la ciudad de origen está previamente introducida.

Opción 5: (*2 puntos*) Pedirá al usuario el nombre de una ciudad origen A, de una ciudad destino B, y mostrará si se puede viajar de A hasta B en dos trayectos; es decir, si hay una línea desde A hasta otra ciudad cualquiera, y otra desde esta última hasta B, indicando el nombre de la ciudad intermedia. Si hay más de una posibilidad, es indiferente cuál se muestre. Si no hay “doble trayecto”, se mostrará un mensaje indicándolo. Podemos suponer que el usuario no va a cometer errores, y las ciudades A y B están previamente introducidas.

NO podemos asumir que se introducen otros datos que no sean los que se indican aquí. Se introducirán obligatoriamente en el orden en que se indica. El modelo de datos se debe implementar con tablas; no se pueden usar EDD.

- El fichero fuente .java a entregar debe compilar sin errores (si no compila un ejercicio su nota máxima es 4 sobre 10).

- No se deben producir excepciones (por cada situación distinta que produzca una excepción, se restará 1 punto de la nota final obtenida).

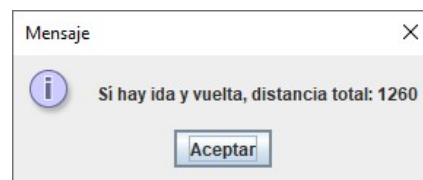
- Sólo se puede presuponer alguna condición si no contradice el enunciado (en caso de contradecirlo, el apartado tendrá 0 puntos). Se valorará el código correcto, indentado y comentado; la claridad de código y su eficiencia y la inexistencia de código o variables superfluas.

EJEMPLOS DE DATOS DE ENTRADA:

Con los datos de entrada siguientes (que son los que mostraría el apartado 2), los resultados deben ser:



El apartado 3, si se introducen “Madrid” y “Barcelona” mostraría:



El apartado 4, si introducimos “Zaragoza” y 400Kms., mostraría:



El apartado 5, si introducimos “Barcelona” y “Sevilla”, mostraría:

