

UD02.EXAME Práctico

DAM1-Programación 2023-24
2023/11/08

1. Coste de viaje en coche (2,5)	2
2. Minutos transcurridos (2,5)	3
3. Gálibo (2,5)	4
4. ViajeCompartido (2,5)	5

- Puedes utilizar apuntes y materiales que consideres pero deberás realizar los programas individualmente. En caso contrario se retirará el examen.
- Realiza programas bien estructurados, legibles, con comentarios, líneas en blanco, identificadores adecuados, etc.
- Cuida la interacción con el usuario, presentando la información de forma clara y ordenada.
- Utiliza los casos de prueba de ejemplo para probar tu programa y definir la salida por pantalla.

- Crea un **paquete** de nombre **examenud02** donde agrupar los ficheros de código fuente. Si es necesario crea un nuevo proyecto/carpeta Java.
- Crea dentro del paquete ficheros **.java** por cada ejercicio con el nombre indicado.
- Incluye al inicio de cada programa un **comentario con tu nombre y apellidos**.
- **Comprime y entrega la carpeta del paquete en el Aula Virtual al terminar.**
- **Tiempo estimado: 2 horas**

1. Coste de viaje en coche (2,5)

CosteViaje.java

Crea una función que calcula el importe en euros de un viaje en coche en función de los kilómetros del trayecto, el tipo de combustible utilizado (0 = Gasolina, 1 = Diesel, 2 = Electricidad) y el consumo medio por cada 100 km, litros (l) para coches gasolina y diesel o Kilovatios/Hora (kWh) para coches eléctricos.

Considera en los cálculos de la función los siguientes precios medios en euros de los distintos combustibles, según <https://www.dieselogasolina.com/> y <https://tarifaluzhora.es>:

- Gasolina, Sin plomo 95: 1.638 € / litro
- Diesel, Gasóleo A: 1.638 € / litro
- Electricidad, kWh: 0.1692 € / kWh

Utiliza el siguiente prototipo implementar para la función.

```
static double costeViaje (double km, byte tipoCombustible, double consumoMedio);
```

Por ejemplo, para un trayecto de 100km,

- Un coche de gasolina que consuma 6 litros a los 100 km gastará: 9,828 €
- Un coche diesel que consuma 5 litros a los 100 km gastará: 8,19 €
- Un coche eléctrico que consuma 15 kWh a los 100 km gastará: 2,538 €

2. Minutos transcurridos (2,5)

MinutosTranscurridos.java

Implementa una función que calcule los minutos transcurridos entre dos instantes de tiempo del mismo día, A y B, a partir del número de hora y minuto de cada instante:

```
static int minutosTranscurridos(int horaA, int minutoA, int horaB, int minutoB);
```

Ten en cuenta que cualquiera de los dos instantes de tiempo puede ser el mayor o incluso que pueden ser iguales. En cualquiera de estos casos el valor devuelto por la función será positivo o cero.

Se espera que las horas de entrada estén entre 0 y 23 y los minutos entre 0 y 59, pero deberás validarlos. Consnidera definir una nueva función para validar los datos de entrada y evitar repetir código. Si algún valor de entrada no es correcto la función devolverá -1.

Puedes utilizar el siguiente conjunto de casos de prueba para depurar tu función:

```
// minutosTranscurridos(13,20,10,10) == 190
// minutosTranscurridos(13,10,10,20) == 170
// minutosTranscurridos(10,10,13,20) == 190
// minutosTranscurridos(10,20,13,10) == 170
// minutosTranscurridos(11,20,10,10) == 70
// minutosTranscurridos(11,10,10,20) == 50
// minutosTranscurridos(10,10,11,20) == 70
// minutosTranscurridos(10,20,11,10) == 50
// minutosTranscurridos(10,10,10,10) == 0
// minutosTranscurridos(10,20,10,10) == 10
// minutosTranscurridos(10,10,10,20) == 10
// minutosTranscurridos(24,10,10,20) == -1
// minutosTranscurridos(10,60,10,20) == -1
// minutosTranscurridos(10,10,24,20) == -1
// minutosTranscurridos(10,10,10,60) == -1
```

3. Gálibo (2,5)

Galibo.java

Se conoce como el gálibo a la dimensión máxima que debe tener un vehículo para poder pasar por una zona de una vía de tamaño reducido como un túnel o un puente.

En carretera el gálibo del que hay que preocuparse normalmente suele ser el vertical, es decir, la altura libre por debajo del puente que define la altura máxima de los vehículos que podrán atravesarlo.

Para evitar problemas con los vehículos más grandes en las carreteras se colocan carteles y otros mecanismos para indicar a los conductores la altura de puente próximo. Gracias a esta información se pueden registrar las alturas de los puentes de un trayecto para conocer si el vehículo grande podrá pasar por debajo de todos ellos.

Escribe un programa que lea el número de puentes o túneles (hasta 100) que hay en el trayecto que se desea hacer. A continuación, para cada uno se leerá de teclado el número de carriles que pasan por debajo (como mucho 5) y la altura de cada uno de los carriles en centímetros (un número entre 200 y 800).

Para terminar, el programa escribirá en pantalla la altura máxima que puede tener el camión para poder realizar el trayecto, teniendo en cuenta que este puede utilizar cualquiera de los carriles disponibles en cada puente/túnel.

Ejemplos de entradas y salidas

	Ejemplo 1	Ejemplo 2	Ejemplo 3	Ejemplo 4
ENTRADA	2 3 300 300 250 2 325 300	2 1 450 1 400	3 1 400 2 350 300 3 200 300 450	1 3 300 350 400
SALIDA	300	400	350	400

4. ViajeCompartido (2,5)

ViajeCompartido.java

Escribe un programa que simule una sencilla aplicación de viajes en coche compartido. El programa debe pedir al usuario que introduzca:

1. El origen y destino del viaje
2. La distancia en kilómetros a recorrer (debe ser mayor o igual a 1 y puede tener decimales)
3. El número de plazas disponibles para pasajeros (un número entero entre 1 y 4)
4. El tipo de combustible (0 = Gasolina, 1 = Diesel, 2 = Electricidad)
5. El consumo medio del vehículo (litros o kWh según el tipo de combustible)

A continuación, el programa debe mostrar el coste total del viaje, utilizando la función **costeViaje()**, y preguntar al usuario el precio por plaza en euros, que deberá ser mayor que cero y no podrá exceder de los 5 céntimos de euro por kilómetro. El programa puede informar al usuario del precio máximo permitido por plaza y solicitará repetir la entrada si no se ajusta a lo permitido.

Para terminar, el programa solicitará el número de pasajeros (que deberá estar entre 0 y el número de plazas disponibles) y mostrará el total en euros que ingresará el conductor por el viaje.

El programa debe **validar** los datos introducidos por el usuario y solicitar que repita la entrada de datos si no son correctos.

El programa imprimirá los **importes en euros con dos decimales** y, en general, cualquier otro número no entero.