

Programación en Lenguaje Java

Práctica 5.1. Cálculo de escapatorias en un circuito de Fórmula 1



Michael González Harbour
Mario Aldea Rivas

Departamento de Matemáticas,
Estadística y Computación

Este tema se publica bajo Licencia:

[Creative Commons BY-NC-SA 4.0](https://creativecommons.org/licenses/by-nc-sa/4.0/)

Práctica 5-1: Cálculo de escapatorias en un circuito de fórmula 1

Objetivos: Practicar con la entrada/salida

Descripción: En un circuito de fórmula 1 a la salida de cada curva se sitúan zonas de escapatoria en las que el coche puede frenar con seguridad si no consigue tomar la curva



Crearemos una clase para calcular estas escapatorias y un main para introducir los datos y mostrar resultados

Diseño de la clase Coche

La clase **Coche** almacena como atributos los coeficientes de frenado y el nombre de la escudería:

- **alfa**: coeficiente de la deceleración aerodinámica en s^{-1}
- **beta**: coeficiente de la deceleración por rozamiento con el suelo en m/s^2
- **escuderia**: nombre de la escudería

Dispone de operaciones para calcular la distancia recorrida, el tiempo de frenado y la longitud de la escapatoria

Coche
-double alfa, beta -String escuderia
+Coche(double alfa, beta, String escuderia) +double distancia(double t, double v0) +double tiempoFrenado(double v0) +double longitudEscapatoria(double v0)

Detalles sobre la clase

- Atributos y otros datos
 - Usaremos unidades del sistema internacional
 - Documentar con comentarios las unidades usadas en los atributos, los argumentos de los métodos, y los valores retornados por ellos
- *Constructor*
 - pone los valores iniciales de los coeficientes y de la escudería, a partir de los parámetros
- *distancia*
 - retorna la distancia recorrida, x , en función del tiempo, t , y de la velocidad inicial, v_0 :

$$x = \frac{1}{\alpha} \left(v_0 + \frac{\beta}{\alpha} \right) (1 - e^{-\alpha t}) - \frac{\beta}{\alpha} t$$

Detalles sobre la clase (cont.)

- tiempoFrenado

- retorna el tiempo hasta detenerse, t_f , en función de la velocidad inicial, v_0 :

$$t_f = \frac{1}{\alpha} \ln\left(\frac{\alpha v_0 + \beta}{\beta}\right)$$

- longitudEscapatoria

- retorna la longitud de la escapatoria en función de la velocidad inicial, v_0
- se calcula con los métodos anteriores como la distancia recorrida en un tiempo t igual al tiempo de frenado

Programa principal

En una clase aparte crear un método `main` que lea de teclado los datos del coche, usando una ventana de la clase `Lectura` del paquete `fundamentos`, y cree un objeto de la clase `Coche` con los datos leídos

Pueden usarse datos de una de las filas de esta tabla:

Coche	alfa (s^{-1})	beta (m/s^2)
Ferrari	0.18	8.9
Red Bull	0.20	8.7

Posteriormente el `main` debe mostrar en pantalla los tiempos de frenado del coche en función de tres velocidades iniciales

- la salida tendrá un formato similar al ejemplo que aparece a continuación

Programa principal (cont.)

```
Tiempos de frenado (s)
Vel(km/h)           Ferrari
100.0 2.4768753522141775
200.0 X.XXXXXXXXXXXXXXXXXX
350.0 6.0406262967454225
```

Finalmente debe mostrar en pantalla las longitudes de la escapatoria para el coche en función de las tres velocidades iniciales anteriores, con este formato aproximado:

```
Longitudes de escapatorias (m)
Vel(km/h)           Ferrari
100.0 31.853261905953275
200.0 XXX.XXXXXXXXXXXXXXXXXX
350.0 241.4480454510441
```

Nota: X.XXXXXXXXXX se sustituirá por el número obtenido

Informe

Entregar:

- El código java de la clase **Coche**
- El código Java de la clase principal
- Una captura de pantalla que muestre los resultados de ejecutar el **main** incluyendo la ventana de la clase **Lectura**

Parte avanzada

Descripción: Añadir un nuevo método y modificar el `main`:

- Añadir a la clase `Coche` un nuevo método para calcular la velocidad en función del tiempo y de la velocidad inicial:

$$v = \frac{v_0}{e^{\alpha t}} - \frac{\beta}{\alpha} + \frac{\beta}{\alpha e^{\alpha t}}$$

- Añadir instrucciones al `main` para mostrar la velocidad del coche transcurrida la mitad del tiempo de frenado, para las tres velocidades iniciales utilizadas anteriormente
- Modificar el `main` para mostrar los resultados en una ventana de la clase `Escritura`

Informe de la parte avanzada

Entregar, además de lo anterior:

- Código Java del nuevo método
- Código Java de la nueva clase principal
- Una captura de pantalla resultante de ejecutar el nuevo `main`