

Recuperatorio - Algoritmos I Taller: Tema A

Recuperatorio Parcial 1 - Funcional

Debés entregar el código completo en los campos correspondientes de cada ejercicio del formulario en el que completaste tus datos personales. Este código debe poder ejecutarse en haskell sin errores. Te recomendamos para ello que pruebes con diferentes ejemplos antes de entregar.

Sos un apasionado de los deportes que involucren cualquier tipo de motor. Cada espectáculo motor puede ser automovilismo o motociclismo. Además para organizarte y perderte lo menos posible disponés de un listado de los distintos eventos que ocurren en un fin de semana.

Ejercicio 1:

Definí el tipo `EspectaculoMotor` que consta de dos constructores `Automovilismo` y `Motociclismo` con los siguientes parámetros:

- El constructor `Automovilismo` debe tomar como parámetros el nombre de la categoría, nombre circuito, horario de comienzo (hora) y duración estimada (en minutos).
- El constructor `Motociclismo` debe tomar como parámetros el nombre de la categoría, horario de comienzo (hora) y cantidad de vueltas.

Una vez que definas el tipo, definí los siguientes términos:

```
formula1 :: EspectaculoMotor
```

```
formula1 = <COMPLETAR>
```

correspondiente al espectáculo de automovilismo "Formula1" que corre en "Singapur" empezando a las 9hs y con una duración aproximada de 120 minutos.

```
motoGP :: EspectaculoMotor
```

```
motoGP = <COMPLETAR>
```

correspondiente al espectáculo de motociclismo "MotoGP" con horario de comienzo 11hs y que correrán 30 vueltas.

Ejercicio 2:

Definí la función `carreraLarga :: EspectaculoMotor -> Bool` que dado un `EspectaculoMotor`, devuelve `True` si la carrera de automovilismo tiene una duración de mayor o igual a 180 minutos, `False` caso contrario.

Ejercicio 3:

Definí la función

```
findeCat :: [EspectaculoMotor] -> String -> [EspectaculoMotor]
```

que dado tu listado de espectáculos motor `fm`, y el nombre de una categoría `cat`, devuelve un nuevo listado que tiene solamente los espectáculos con nombre de categoría `cat` (en el mismo orden que aparecen en `fm`).