

## Autos Locos

Todos los fines de semana tiene lugar la competencia que apasiona a multitudes: la carrera de los Autos Locos, como últimamente este evento se ha visto afectado por condiciones sospechosas los organizadores han solicitado su ayuda para intentar develar los misterios que se esconden en este emocionante evento.

Para satisfacer los requerimientos del cliente se ha optado por la confección de un programa Prolog capaz de responder las inquietudes de los organizadores, y usted ha sido seleccionado como el programador estrella del proyecto.

Para esto se definieron los siguientes hechos que usted deberá utilizar.

- Una carrera tiene un nombre y muchos tramos, donde cada tramo tiene la peculiaridad de poseer una característica especial:

```
%carrera(nombre de la carrera, lista de tramos que la componen)
carrera(villaCalibre, [tramo(ciudad),tramo(carretera),tramo(laguna)]).
carrera(adelaida, [tramo(montania),tramo(bosque),tramo(trampolín),tramo(montania)]).
```

- Las características especiales de los tramos tienen obstáculos que son propios de él, por ejemplo:

```
%obstáculo(característica del terreno, obstáculos que se presentan)
obstáculo(montania, barro).
obstáculo(montania, ripio).
obstáculo(laguna, agua).
obstáculo(laguna, animalSalvaje).
obstáculo(bosque, pasto).
obstáculo(trampolín, aire).
```

(carreteras y ciudades no tienen obstáculos).

- Un obstáculo puede requerir de capacidades especiales para ser sorteado

```
%capacidadRequerida(obstaculo,capacidad para superarlo)
capacidadRequerida(barro, todoTerreno).
capacidadRequerida(ripio, estabilidad).
capacidadRequerida(agua, waterResistant).
capacidadRequerida(animalSalvaje, _). %cualquier capacidad puede con un animal salvaje
capacidadRequerida(pasto, todoTerreno).
capacidadRequerida(aire, volar).
```

- Cada auto tiene características especiales, por ejemplo algunos autos pueden volar, otros transformarse en muchas cosas, otros defenderse con distintos dispositivos, otros ir mas rapido. Las características especiales se representan como funtores y pueden ser:

- transformacion(resultado de la transformacion)
- capacidad(capacidad)

```
feature(superFerrari, arma(amestralladora)).
feature(superFerrari, arma(canon)).
feature(superFerrari, transformacion(barco)).
feature(superFerrari, capacidad(velocidad)).
```

```
feature(elRocomovil, capacidad(velocidad)).
feature(elRocomovil, capacidad(estabilidad)).
```

```
feature(elStukaRakuda, capacidad(volar)).
feature(elStukaRakuda, capacidad(velocidad)).
feature(elStukaRakuda, arma(amestralladora)).
```

```
feature(superConvertible, transformacion(barco)).
feature(superConvertible, transformacion(bicicleta)).
feature(superConvertible, transformacion(dirigible)).
feature(superConvertible, transformacion(tanque)).
```

```
feature(compactPussycat, capacidad(velocidad)).
feature(compactPussycat, arma(cosmeticos)).
```

- Cada auto tiene competidores que viajan en el:  
%equipo(Auto,Miembros) siendo el primero de los miembros el conductor  
equipo(superFerrari, [pierNodoyuna, patan]).  
equipo(elRocomovil, [piedroMacana, rocoMacana]).  
equipo(elStukaRakuda, [baronHansFritz]).  
equipo(superConvertible, [profesorLocobitch]).  
equipo(compactPussycat, [penelopeGlamour]).  
equipo(alambiqueVeloz, [lucasElGranjero, elOsoMiedoso]).

Se pide:

1. Según las capacidades del auto
  - a. esAutoPeligroso\1 si tiene mas de dos armas o patan viaja en él.
  - b. vaConPropulsion\1 un auto va con propulsión si alguno de sus capacidades es la velocidad.
  - c. esSeguroEstarCerca\1 cuando un auto no tiene armas siempre es seguro estar cerca.

2. puedeSortearObstaculo\2, un auto puede sortear un obstáculo si tiene como feature la capacidad requerida por el obstáculo para ser superado. Cualquier feature sirve para superar a un animal salvaje.

Por ejemplo:

```
?- puedeSortearObstaculo(Auto,aire).  
Auto = superConvertible ;  
Auto = elRocomovil ;  
Porque el superConvertible se transforma en un dirigible que puede volar y  
elRocomovil cuenta con la ayuda animal de un terodactilo que lo deja volar.
```

```
?- puedeSortearObstaculo(compactPussycat,Obstaculo).  
Obstaculo = animalSalvaje ;  
Esto es cierto porque el compact pussycat tiene al menos una caracteristica  
especial (de hecho tiene dos, asi que le sobran y todo)
```

3. autoConclusiva\1 una carrera es autoconclusiva si el último tramo y el primer tramo que la componen son el mismo terreno. Este predicado debe ser inversible.

Por ejemplo:

```
?- autoConclusiva(Carrera).  
Carrera = adelaída.
```

4. puedeTransitar\2 que relaciona a una carrera y un automóvil si ese automóvil puede sortear todos los obstáculos que aparecen en cada uno de los tramos de la carrera. Este predicado tiene que ser totalmente inversible.
5. esImpochible\1 se dice que una carrera es imposible si no hay ningún automóvil que sea capaz de transitarla. Debe ser inversible.
6. carreraComplicada/1 Una carrera es complicada si tiene un paso complicado entre dos de sus tramos (obviamente hablamos sólo de tramos consecutivos).  
Un paso desde un tramo a otro es complicado si
  - a. se pasa de carretera a montania o,
  - b. se pasa de cualquier cosa a laguna

Hint: tal vez convenga hacer un predicado auxiliar pasoComplicado/2.