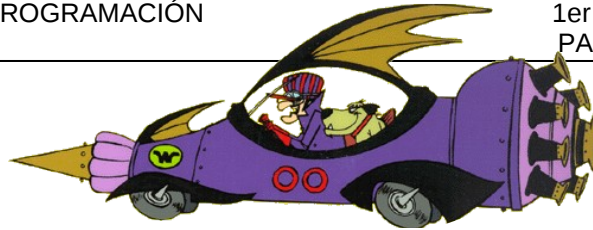


Autos



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(montania),tramo(laguna)]).
carrera(casaCristo,[tramo(montania),tramo(laguna),tramo(laguna),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).           %cualquier capacidad puede con un animal
capacidadRequerida(ripio, estabilidad).           salvaje
capacidadRequerida(agua, waterResistant).        capacidadRequerida(animalSalvaje, _).
                                                    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, transformacion(barco)).
feature(superFerrari, capacidad(velocidad)).
feature(superFerrari, trampa(ametralladora)).
feature(superFerrari, trampa(aceite)).
```

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

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

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

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

```
feature(compactPussycat, trampa(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]).
```

- A su vez, los competidores tienen un nivel de aptitud, dependiendo de su personalidad:

```
%nivelAptitud(personalidad, nivel).
nivelAptitud(malhechor, 35).
nivelAptitud(servicial, 20).
nivelAptitud(propensoAccidentes, 10).
nivelAptitud(damiselaEnApuros, 10).
nivelAptitud(nadaParaDestacar, 5).
```

Se pide:

1. personalidad\2 que relaciona a un competidor con su tipo de personalidad. Las características se determinan así:
 - a. malhechor, un competidor es un malhechor si viaja en el superFerrari o viaja en un automóvil que esta equipado con dos trampas.
 - b. servicial un competidor es servicial si no es un malhechor y puede llevar algún pasajero extra. Los que pueden llevar pasajeros extras son:
 - los que pueden transformar su vehiculo en una bicicleta
 - los que viajan solos
 - c. propensoAccidentes un competidor es propenso a accidentes si su vehiculo no puede transformarse.
 - d. damiselaEnApuros un competidor es una damisela en apuros si es propenso a accidentes y viaja solo.

Con la base de conocimiento actual se puede deducir qué:

- pierNodoyuna y patan son malhechores
- baronHansFritz, profesorLocobitch y penelopeGlamour son serviciales
- piedroMacana, rocoMacana, baronHansFritz, penelopeGlamour, lucasElGranjero y elOsoMiedoso son propensos a accidentes
- penelopeGlamour y el baronHansFritz (JA!) son damiselas en apuros

2. puedeSortearObstáculo\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 = elStukaRakuda;
```

Porque el elStukaRakuda se tiene la capacidad de volar

```
?- puedeSortearObstaculo(compactPussycat,Obstaculo).
Obstaculo = animalSalvaje ;
```

Esto es cierto porque el compact pussycat tiene al menos una característica especial (de hecho tiene dos, así 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 = casaCristo.
```

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. esImpoachable\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
- se pasa de carretera a montaña o,
 - se pasa de cualquier cosa a laguna

Con la base de conocimiento actual `villaCalibre` y `casaCristo` son carreras complicadas.

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

7. Nivel de aptitud

- a. `nivelDeAptitudDelCompetidor\2`: esta dado por las personalidades del competidor. Como un competidor puede tener más de una personalidad, su nivel de aptitud es la suma del nivel de aptitud de todas sus personalidades.

?- `nivelDeAptitudDelCompetidor(penelopeGlamour, Total)`.
Total = 40

?- `nivelDeAptitudDelCompetidor(pierNodoyuna, Total)`.
Total = 35

?- `nivelDeAptitudDelCompetidor(piedroMacana, Total)`.
Total = 10

- b. `nivelDeAptitudDelVehiculo\2`: es la suma de los puntos que le dan cada feature, siendo:

- 5 por cada trampa
- 10 por cada transformación
- 15 por cualquier otro feature

?- `nivelDeAptitudDelVehiculo(superFerrari, Total)`.
Total = 35

?- `nivelDeAptitudDelVehiculo(compactPussycat, Total)`.
Total = 20

?- `nivelDeAptitudDelVehiculo(elStukaRakuda, Total)`.
Total = 30

- c. `nivelDeAptitudParaCompetir\2` relaciona a un vehiculo con su nivel de aptitud para una competencia. Esto es la sumatoria del nivel de aptitud de todos los competidores que viajan en el vehiculo más el nivel de aptitud del vehiculo.

?- `nivelDeAptitudParaCompetir(superFerrari, Total)`.
Total = 105

?- `nivelDeAptitudParaCompetir(compactPussycat, Total)`.
Total = 60

?- `nivelDeAptitudParaCompetir(elStukaRakuda, Total)`.
Total = 70

8. Saber qué vehículo llega primero. Es el vehículo con el mayor nivel de aptitud y que además no es el rocomovil.

Contra todo pronóstico, con está base de conocimiento, el vehículo que llega primero es el superFerrari.