

El Parque de la Costa está por implementar un nuevo sistema de asociados, y por ende lo contrata a Ud para que resuelva ciertas consultas mediante predicados en Prolog.

La base de conocimientos cuenta con:

El predicado entrada/3 que relaciona el nombre de una entrada, su precio, y una lista con los juegos a los que permite acceder, por ejemplo:

```
entrada(socio(comun),10,[sillasVoladoras,autitosChocadores,elPulpo,...]).
entrada(socio(especial),15,[sillasVoladoras,autitosChocadores,elPulpo,boomerang,elDesafio,...]).
entrada(socio(gold),25,[sillasVoladoras,autitosChocadores,elPulpo,boomerang,elDesafio,vertigoXtremo...]).
entrada(excepcional,30,[sillasVoladoras,autitosChocadores,elPulpo,elDesafio,vertigoXtremo...]).
entrada(invitado(mayor),40,[sillasVoladoras,autitosChocadores,elPulpo,boomerang,elDesafio,...]).
entrada(invitado(menor),30,[sillasVoladoras,autitosChocadores,elPulpo,...]).
```

El predicado dinero/2 relaciona a una persona con la cantidad de dinero que tiene:

```
dinero(ana,35).
dinero(carlos,40).
dinero(rodrigo,50).
dinero(marcelo,25).
```

El predicado edad/2 relaciona a una persona con su edad. Una persona se considera mayor si tiene 12 años o más:

```
edad(ana,24).
edad(carlos,19).
edad(rodrigo,11).
```

El predicado vertigoTolerable/2 indica mediante un número entero el nivel de vértigo que una persona puede tolerar:

```
vertigoTolerable(ana,60).
vertigoTolerable(carlos,70).
vertigoTolerable(rodrigo,20).
```

El predicado vertigoDeJuego/2 indica mediante un número entero el nivel de vértigo que un juego posee. Existe una cláusula por cada juego existente en el Parque:

```
vertigoDeJuego(elVigia,25).
vertigoDeJuego(elDesafio,40).
vertigoDeJuego(boomerang,60).
```

1) Desarrollar el predicado puedePagar/2 que relaciona una persona con un nombre de entrada, si la persona tiene el dinero para pagar dicha entrada y, si es invitado, tiene la edad requerida. Los socios y entradas excepcionales no tienen restricciones de edad.

?- puedePagar(ana, socio(comun)).

Yes.

?- puedePagar(rodrigo, invitado(mayor)).

No.

(si bien rodrigo tiene dinero suficiente para pagar la entrada, es menor de edad y no puede adquirir la entrada de invitado mayor).

2) Desarrollar el predicado subeACualquierJuego/1 que indica qué personas pueden subir a cualquier juego del Parque. Una persona puede subir a un juego si su vértigo tolerable es mayor o igual que el nivel de vértigo del juego. El predicado debe ser inversible.

?- subeACualquierJuego(rodrigo).

No.

?- subeACualquierJuego(X).

X = ana;

X = carlos;

No.

3) Desarrollar el predicado cualesContienen/2 que relaciona un juego con una lista de entradas que permiten acceder a dicho juego, por ejemplo:

?- cualesContienen(vertigoXtremo, Entradas).

Entradas = [socio(gold), excepcional];

No.

?- cualesContienen(boomerang, Entradas).

Entradas = [socio(comun), socio(especial), socio(gold)];

No.

?- cualesContienen(Juego, [socio(comun), socio(gold)]).

Juego = sillasVoladoras;

Juego = autitosChocadores;

Juego = elPulpo;

No.

4) Desarrollar el predicado rankingDeMaricotas/1 que recibe una lista de personas y responde Yes si la misma cumple las siguientes condiciones:

- Está ordenada ascendentemente según el nivel de vértigo tolerable.
- El último de la lista soporta el vértigo de subir a cualquier juego.

?- rankingDeMaricotas([rodrigo, ana, carlos]).

Yes.

?- rankingDeMaricotas([ana, rodrigo]).

No.

5) Desarrollar el predicado evitaSerSocio/1 que indica si una persona puede comprar algún abono que no sea de socio.

?- evitaSerSocio(marcelo).

No.