

¿Democracia? ¡Lógico!



Siendo año de elecciones en la Argentina, se nos pidió modelar nuestro sistema electoral en el lenguaje de Prolog. Como en cualquier desarrollo, utilizaremos datos ficticios para el modelado (no habrá similitudes ni referencias a la realidad).

Antes de arrancar... el formato de entrega

Fecha de finalización: El assignment finaliza el miércoles 14/08 a las 18:00 hs. Pasado ese lapso el docente no verá los commits.

El TP será individual. Tenés que estar al tanto de la solución presentada y poder defenderla, lo que significa justificar las decisiones de diseño tomadas para resolver los casos de uso planteados.

Defensa: la defensa del TP será presencial, el mismo miércoles 14/08 a partir de las 18:30. En caso de enfermedad o causa de fuerza mayor, debe enviar un mail a los tutores avisando el motivo de la ausencia y sugiriendo una fecha para defender el TP.

Qué debe incluir en el repositorio: el código fuente en Prolog que resuelva correctamente los puntos que se describen a continuación, la implementación de los casos de prueba exitosos (recomendamos hacer un `begin_tests` por cada punto), y en el README los integrantes del equipo y el *badge* contra Travis con el estado del último *build*. Todos los puntos anteriores más la aprobación de la defensa del TP permitirá que puedan presentarse a rendir el primer examen del Paradigma Lógico, de acuerdo a las reglas de juego que se comunicaron al principio del cuatrimestre.

1 Candidatos y partidos

Contamos con la siguiente información de los candidatos:

- Frank es candidato del partido Rojo
- Claire es candidata del partido Rojo
- Garrett es candidato del partido Azul
- Frank tiene 50 años
- Claire tiene 52 años
- Garrett tiene 64 años
- Peter no es candidato del partido Amarillo y tiene 26 años
- Jackie es candidata del partido Amarillo y tiene 38 años
- El partido violeta no tiene candidatos
- Linda es candidata del partido Azul
- Linda tiene 30 años
- Catherine es candidata del partido Rojo y tiene 59 años
- Seth es candidato del partido Amarillo
- Heather es candidata del partido Amarillo, nació en el año 1969

Además, sabemos en qué provincias se postulan los partidos:

- El partido azul se postula en Buenos Aires, Chaco, Tierra del Fuego, San Luis y Neuquén.
- El partido rojo se presenta en Buenos Aires, Santa Fe, Córdoba, Chubut, Tierra del Fuego y San Luis. Finalmente no se presentará en Formosa.
- El partido amarillo se presenta en Chaco, Formosa, Tucumán, Salta, Santa cruz, La Pampa, Corrientes, Misiones y Buenos Aires.

Adicionalmente, contamos con la información de la cantidad de habitantes de cada provincia:

Provincia	Cantidad de habitantes
Buenos Aires	15.355.000
Chaco	1.143.201
Tierra del Fuego	160.720
San Luis	489.255
Neuquén	637.913

Santa Fe	3.397.532
Córdoba	3.567.654
Chubut	577.466
Formosa	527.895
Tucumán	1.687.305
Salta	1.333.365
Santa Cruz	273.964
La Pampa	349.299
Corrientes	992.595
Misiones	1.189.446

Definir la base de conocimientos. En caso de que algo no se refleje en la misma justificar el por qué.

En este punto no hace falta implementar testeo unitario, se pide que identifique dónde aparece el Principio de Universo Cerrado y justifique su decisión.

2 Provincia picante

Queremos saber si una provincia **esPicante/1**. Esto se da cuando al menos dos partidos presentan candidatos en dicha provincia y además ésta tiene más de 1 millón de habitantes.

Por ejemplo:

- Buenos Aires es picante por tener 15 millones de habitantes y los partidos azul, rojo y amarillo presentan candidatos.
- Santa Fe no es picante, tiene 1 millón de habitantes pero solo un partido presenta candidatos
- San Luis no es picante. Varios presentan candidatos pero tiene 400 mil habitantes.

Construir los casos de prueba dando nombres representativos a cada uno de los tests (recordar que en lugar de *santa_fe_no_es_picante*, es mejor decir *provincia_no_picante_por_un_partido_solo*). No repetir ejemplos que estén en la misma clase de equivalencia.

3 PASO

Contamos con la intención de voto de cada partido en cada provincia (en porcentajes):

```
% intencionDeVotoEn(Provincia, Partido, Porcentaje)
intencionDeVotoEn(buenosAires, rojo, 20).
intencionDeVotoEn(buenosAires, azul, 40).
intencionDeVotoEn(buenosAires, amarillo, 40).
intencionDeVotoEn(chaco, rojo, 50).
intencionDeVotoEn(chaco, azul, 20).
intencionDeVotoEn(chaco, amarillo, 0).
intencionDeVotoEn(tierraDelFuego, rojo, 40).
intencionDeVotoEn(tierraDelFuego, azul, 20).
intencionDeVotoEn(tierraDelFuego, amarillo, 10).
intencionDeVotoEn(sanLuis, rojo, 50).
intencionDeVotoEn(sanLuis, azul, 20).
intencionDeVotoEn(sanLuis, amarillo, 0).
intencionDeVotoEn(neuquen, rojo, 80).
intencionDeVotoEn(neuquen, azul, 10).
intencionDeVotoEn(neuquen, amarillo, 0).
intencionDeVotoEn(santaFe, rojo, 20).
intencionDeVotoEn(santaFe, azul, 41).
intencionDeVotoEn(santaFe, amarillo, 39).
intencionDeVotoEn(cordoba, rojo, 10).
intencionDeVotoEn(cordoba, azul, 60).
intencionDeVotoEn(cordoba, amarillo, 20).
intencionDeVotoEn(chubut, rojo, 15).
intencionDeVotoEn(chubut, azul, 15).
intencionDeVotoEn(chubut, amarillo, 15).
intencionDeVotoEn(formosa, rojo, 0).
intencionDeVotoEn(formosa, azul, 0).
intencionDeVotoEn(formosa, amarillo, 0).
intencionDeVotoEn(tucuman, rojo, 40).
intencionDeVotoEn(tucuman, azul, 40).
intencionDeVotoEn(tucuman, amarillo, 20).
intencionDeVotoEn(salta, rojo, 30).
intencionDeVotoEn(salta, azul, 60).
```

```
intencionDeVotoEn(salta, amarillo, 10).
intencionDeVotoEn(santaCruz, rojo, 10).
intencionDeVotoEn(santaCruz, azul, 20).
intencionDeVotoEn(santaCruz, amarillo, 30).
intencionDeVotoEn(laPampa, rojo, 25).
intencionDeVotoEn(laPampa, azul, 25).
intencionDeVotoEn(laPampa, amarillo, 40).
intencionDeVotoEn(corrientes, rojo, 30).
intencionDeVotoEn(corrientes, azul, 30).
intencionDeVotoEn(corrientes, amarillo, 10).
intencionDeVotoEn(misiones, rojo, 90).
intencionDeVotoEn(misiones, azul, 0).
intencionDeVotoEn(misiones, amarillo, 0).
```

Modelar el predicado **leGanaA/3** el cual relaciona a dos candidatos y una provincia, y nos dice si un candidato le ganaría a otro en una provincia. Para ello:

- El partido del ganador debe competir en la provincia.
- Si el partido del perdedor también compite en la provincia, se evalúa el que tenga el mayor porcentaje de votos en la provincia. Si hay empate, no se cumple la relación.
- Si ambos candidatos pertenecen al mismo partido, la relación se cumple si el partido compite en la provincia.

Por ejemplo:

```
> leGanaA(frank, garrett, tierraDelFuego)
True
% Ambos partidos compiten en Tierra del Fuego, y el rojo tiene más
votos que el azul.

> leGanaA(frank, jackie, santaFe)
True
% El partido de Frank es el único que se postula en Santa Fe, por ende
es el ganador.

> leGanaA(claire, jackie, misiones)
False
% El partido de Claire no se presenta en Misiones, por ende no se
cumple la relación.

> leGanaA(frank, claire, tierraDelFuego)
```

True

% Frank le gana a Claire por ser del mismo partido y presentarse en la provincia.

> leGanaA(heather, linda, buenosAires)

False

% empatan en votos

Implemente los 5 casos de prueba, eligiendo nombres representativos para los tests.

4 Partido hegemónico

Se pide resolver **partidoHegemonico/1**. Un partido es hegemónico si es favorito en las provincias consideradas importantes, que son Buenos Aires, Córdoba y Santa Fe. Un partido es favorito en una provincia si todos los demás partidos tienen menos intención de voto que el partido ganador. Debe resolver este punto sin utilizar findall/3, recomendamos trabajar con el predicado forall/2.

Implemente los casos de prueba, ¿cuáles son los casos de equivalencia que surgen? ¿Cómo podemos testear los que no son partidos hegemónicos, qué conceptos aparecen?

5 Promesas

Durante la campaña los candidatos hacen tres tipos de promesa:

- que la inflación se mantenga en un determinado porcentaje **anual**
- construir una lista de cosas
- o aumentar una cantidad de puestos de trabajo

Las promesas pueden ser realizables o no, esto se determina de diferente manera:

- mantener la inflación por debajo de 10% no es realizable, de lo contrario sí
- se podrían construir menos de 3 cosas, siempre y cuando una de esas cosas no sea una universidad. Esto significa: si el candidato dice que entre esas cosas va a construir una universidad, no es realizable, de lo contrario la promesa se puede cumplir si se trata de menos de 3 cosas.
- se pueden aumentar tantos puestos de trabajo como (la edad del candidato * 10.000). Por ejemplo, Frank puede prometer hasta 500.000 puestos de trabajo = 50 es su edad * 10.000, más de eso no es realizable

Queremos saber cuáles son los candidatos serios, que son los que hacen al menos 3 promesas realizables. Este punto es el único que recomendamos resolver con `findall/3`.

Por ejemplo:

```
promete(garrett, construir([escuela, hospital, universidad, ruta])).
promete(garrett, metaDeInflacion(18)).
promete(garrett, construir([casas, autopistas])).
promete(seth, metaDeInflacion(10)).
promete(claire, nuevosPuestosDeTrabajo(200000)).
promete(claire, nuevosPuestosDeTrabajo(520000)).
promete(claire, nuevosPuestosDeTrabajo(1000000)).
promete(claire, metaDeInflacion(4)).
promete(claire, metaDeInflacion(14)).
promete(frank, metaDeInflacion(3)).
promete(frank, metaDeInflacion(13)).
promete(frank, nuevosPuestosDeTrabajo(10000)).
promete(frank, construir([hospital, universidad])).
promete(frank, construir([plazas, autopistas, extensionSubte,
repavimentacionTotal])).
```

Queremos que el predicado sea inversible y además que implemente los siguientes casos de prueba, de la manera en que crea conveniente:

- Garrett no es serio (de las 3 promesas, solo 2 son realizables, la primera incluye una universidad)
- Seth no es serio (la única promesa la puede cumplir pero no llega a 3)
- Claire es seria (de las 5 promesas, 3 son realizables, salvo la inflación del 4% y el millón de puestos de trabajo, el de 520000 cumple con lo justo -caso borde-)
- Frank no es serio (de las 5 promesas, solo 2 son válidas: la meta de inflación del 13% > 10%, y los puestos de trabajo 10.000 muy por debajo de su edad * 10.000; en el resto promete construir muchas cosas, en la otra promete construir una universidad, y la meta de inflación del 3% es incumplible)

6 “Pancakes”

Saber si un candidato es panquequeable, esto se puede dar si el candidato

- es amigo de otro candidato de otro partido
- o bien es amigo de otro candidato del mismo partido, que es panquequeable

Por ejemplo, si tenemos esta lista de amigos:

```
% el primero es amigo del segundo pero no al revés
amigo(frank, claire).
amigo(claire, catherine).
amigo(catherine, jackie).
amigo(jackie, garrett).
amigo(jackie, heather).
amigo(seth, heather).
```

Los panquequeables son:

- Catherine: del partido rojo y amiga de Jackie que es del amarillo
- Jackie: del partido amarillo y amiga de Garrett del azul, por ejemplo
- Frank: amiga de Claire y son del mismo partido, Claire es amiga de Catherine y son ambas del partido rojo, pero Catherine es amiga de Jackie que es del amarillo.
- Claire como veníamos contando antes
- Seth, Garrett, Linda y Heather no son panquequeables

El predicado debe ser inversible y trabajar a n niveles. Implemente los casos de prueba y determine cuáles son las clases de equivalencia que tiene sentido testear en base a los ejemplos planteados (esto significa que se presentaron 4 ejemplos, ud. debe determinar si de esos 4 ejemplos tiene sentido testearlos todos por separado, o agruparlos en diferentes clases).

Temas a evaluar

- Hechos y Reglas
- Universo cerrado
- Negación
- Forall
- Unificación e Inversibilidad
- Functores y Polimorfismo
- Pattern Matching
- Listas
- Findall
- Recursividad
- Testeo unitario