

Práctica 3

La calculadora electoral

Fecha de entrega: 10 de junio de 2016.

Porcentaje en la calificación total de prácticas: 10 % (es decir, 3 % de la nota final de la asignatura).

Esta práctica se divide en dos apartados:

- El primer apartado, que es obligatorio, consiste en la realización de tres consultas de un total de diez que se proporcionan en el enunciado. La elección concreta de las tres consultas a realizar no la deciden los componentes del grupo, sino que viene determinada los DNIs (o NIEs) de los mismos.
- El segundo apartado, que es una extensión opcional, consiste en la implementación de una consulta recursiva de mayor complejidad.

Con el primer apartado la calificación máxima obtenible será de APTO (5/10), que corresponde a 0.15 puntos en la nota final de la asignatura. Para aspirar a la calificación de APTO (10/10, es decir, 0.30 puntos en la nota final) se deberán completar los dos apartados.

Se valorará la corrección y la simplicidad de las reglas implementadas en *Datalog*.

En esta práctica vamos a realizar consultas sobre una BD que contiene los resultados de las pasadas elecciones del 20 de diciembre al congreso de los diputados. Esta BD contiene la cantidad de escaños obtenidos por cada partido en cada circunscripción electoral. Para simplificar el tratamiento, la base de datos se restringe a los partidos que obtuvieron representación parlamentaria (Cuadro 1). La base de datos *Datalog* define una relación `escanyos(C, P, N)`, donde cada fila indica que el partido `P` ha obtenido `N` escaños en la circunscripción electoral `C`. Por ejemplo:

```
escanyos('Granada', 'PSOE', 2).
```

De la lista de partidos mostrada en el Cuadro 1, no todos ellos se han presentado en todas las circunscripciones (por ejemplo, 'EAJ-PNV' no ha concurrido en la circunscripción de Madrid). Las filas correspondientes a estos datos **no** aparecen como hechos en la base de datos. Sin embargo sí se especifican aquellos partidos que, habiéndose presentado en una determinada circunscripción, no obtuvieron ningún escaño en la misma. Por ejemplo:

```
escanyos('Gerona', 'PP', 0).
```

Por otro lado, la base de datos contiene también la distribución de las circunscripciones en las comunidades autónomas. La relación `pertenece(C, CA)` indica que la circunscripción `C` pertenece a la comunidad autónoma `CA`. Por ejemplo:

```
pertenece('Soria', 'Castilla y Leon').
```

Partido	Denominación
Partido Popular	'PP'
Partido Socialista Obrero Español	'PSOE'
Ciudadanos-Partido de la Ciudadanía	'Cs'
Podemos	'PODEMOS'
En Comú Podem	'EN COMU'
Unidad Popular: Izquierda Unida, Unidad Popular en Común	'IU-UPeC'
Compromís-Podemos-És el moment	'PODEMOS-COMPROMIS'
Esquerra Republicana de Catalunya-Catalunya Sí	'ERC-CATSI'
Democràcia i Llibertat. Convergència. Demòcrates. Reagrupament	'DL'
En Marea	'PODEMOS-En Marea-ANOVA-EU'
Euzko Alderdi Jeltzalea-Partido Nacionalista Vasco	'EAJ-PNV'
Euskal Herria Bildu	'EH Bildu'
Coalición Canaria-Partido Nacionalista Canario	'CCa-PNC'

Cuadro 1: Partidos con representación parlamentaria, en orden decreciente de votos obtenidos.

Si una circunscripción no pertenece a ninguna CA, no aparece en la relación [pertenece](#).

En el Campus Virtual encontrarás un fichero *Datalog* con las definiciones de las relaciones mencionadas arriba. A partir de la línea 354 debéis implementar las relaciones que se piden en este enunciado. Al realizar la entrega en el Campus Virtual **no tenéis que subir ningún fichero**; tan solo tenéis que copiar vuestras definiciones en el cuadro de texto que aparecerá en la tarea correspondiente del CV. Esto incluye la definición de la relación [dni](#), que se explicará en el siguiente apartado.

1 Consultas obligatorias (a realizar tres)

En primer lugar, modifica la relación [dni](#) de la BD (líneas 361 y 362) e introduce en cada uno de los hechos de esta relación el DNI (o el NIE, en su caso) de cada componente del grupo, sin incluir la letra¹. Si tu grupo está formado por una única persona, pon tu número de DNI en uno de los hechos de la relación [dni](#) y elimina el otro. A continuación, recarga la base de datos y ejecuta la siguiente consulta en DES:

```
DES> pregunta(N, D)
```

Como resultado de la consulta aparecerán tres filas, indicando las tres consultas que tendrás que realizar de la siguiente lista:

1. Partidos que han obtenido más de seis escaños en alguna circunscripción.

```
partido_fuerte(P) <=> P ha obtenido más de 6 escaños en alguna circunscripción.
```

2. Circunscripciones en las que 'PODEMOS' tiene más de dos escaños.

```
cir_podemos(C) <=> El partido 'PODEMOS' tiene más de dos escaños en C.
```

¹Si no tienes DNI o NIE, utiliza las seis últimas cifras del identificador Moodle en el Campus Virtual.

3. Circunscripciones en las que Ciudadanos ('Cs') tiene más de dos escaños.

`cir_ciudadanos(C) <=> El partido 'Cs' tiene más de dos escaños en C.`

4. Circunscripciones que no pertenecen a ninguna comunidad.

`cir_atomica(C) <=> La circunscripción C no pertenece a ninguna comunidad autónoma.`

5. Comunidades autónomas en las que no se ha presentado Izquierda Unida – Unidad Popular ('IU-UPeC').

`ca_no_iu_upec(CA) <=> El partido 'IU-UPeC' no se ha presentado en ninguna provincia de la comunidad autónoma CA.`

6. Número de circunscripciones por cada comunidad autónoma.

`circunscripciones_ca(CA, N) <=> La comunidad autónoma CA tiene N circunscripciones.`

7. Número de escaños por circunscripción

`escanyo_cir(C, N) <=> A la circunscripción C le corresponden N escaños.`

8. Numero de escanyos por comunidad autonoma.

`escanyo_ca(C, N) <=> A las circunscripciones de la comunidad CA les corresponden N escaños en total.`

9. Número de escaños obtenidos por cada partido en Galicia.

`escanyo_partido(P, N) <=> Al partido P le corresponden N escaños en las circunscripciones de Galicia.`

10. Partidos que, habiéndose presentado en alguna comunidad, no tienen representación parlamentaria en la misma.

`no_representacion(CA, P) <=> El partido P ha obtenido 0 escaños en la comunidad autónoma CA`

2 La calculadora de coaliciones

En esta extensión de la práctica se definirá una relación recursiva que considere las coaliciones que habrían podido formarse en el Congreso de los Diputados. Para ello se definirá una relación que contemple las distintas combinaciones entre los partidos y los escaños que se obtendrían conjuntamente con cada una de ellas.

Antes de comenzar esta parte define una relación `escanyos_total` que contenga, para cada partido, la suma de los escaños obtenidos en las distintas circunscripciones. Esta relación ha de estar definida mediante reglas, y su resultado ha de ser el siguiente:

```
DES> escanyos_total(P, NE)
```

```
{
  escanyos_total('CCa-PNC',1),
  escanyos_total('Cs',40),
  escanyos_total('DL',8),
  escanyos_total('EAJ-PNV',6),
  escanyos_total('EH Bildu',2),
  escanyos_total('EN COMU',12),
  escanyos_total('ERC-CATSI',9),
  escanyos_total('IU-UPeC',2),
  escanyos_total('PODEMOS',42),
  escanyos_total('PODEMOS-COMPROMIS',9),
  escanyos_total('PODEMOS-En Marea-ANOVA-EU',6),
  escanyos_total('PP',123),
  escanyos_total('PSOE',90)
}
```

Info: 13 tuples computed.

Una coalición está formada por un conjunto de partidos. Por generalidad, también consideramos como coaliciones el conjunto vacío y los conjuntos formados por un único partido. Codificaremos cada coalición mediante un número entero, cuya representación binaria indicará qué partidos pertenecen a la coalición y cuales no. Un **1** en la representación binaria indica que el partido correspondiente participa en la coalición, y un **0** indica que no pertenece a dicha coalición.

Dicho lo anterior, tendremos que asignar a cada partido el *bit* correspondiente dentro de la representación binaria de una coalición. Esta asignación viene dada por la relación `partido`, cuya definición se reproduce a continuación:

```
partido('CCa-PNC', 0).
partido('Cs', 1).
partido('DL', 2).
partido('EAJ-PNV', 3).
partido('EH Bildu', 4).
partido('EN COMU', 5).
partido('ERC-CATSI', 6).
partido('IU-UPeC', 7).
partido('PODEMOS', 8).
partido('PODEMOS-COMPROMIS', 9).
partido('PODEMOS-En Marea-ANOVA-EU', 10).
partido('PP', 11).
partido('PSOE', 12).
```

De este modo, por ejemplo, la coalición *PSOE-Podemos-Compromis-En Marea-Ciudadanos* queda codificada mediante el número $(1011100000010)_2 = 2^{12} + 2^{10} + 2^9 + 2^8 + 2^1 = 5890$.

Define una relación recursiva `coalicion(Coalicion, NumEscanyos, NumPartidos)` que asocie cada `Coalicion` (codificada como se ha explicado anteriormente) con la suma total de escaños que obten-

dría (`NumEscanyos`) y el número de partidos que la conforman (`NumPartidos`). Por ejemplo, la coalición anteriormente citada corresponde a la fila `coalicion(5890, 187, 5)`. Algunos consejos:

- Realiza, en el caso recursivo, la llamada recursiva al principio de la regla, con el fin de evitar mensajes de aviso sobre posibles operaciones aritméticas sobre variables no ligadas.

```
% Caso recursivo
coalicion(Coalicion, NumEsc, NumPart) :-
    coalicion(Coalicion1, NumEsc1, NumPart1),
    ... etc ...
```

- Establece dentro del caso recursivo una cota superior al número de escaños, para evitar que DES se quede calculando indefinidamente tuplas de la relación. Una buena cota es `NumEsc =< 350`, que es el número de miembros en el Congreso de los Diputados.
- Igualmente, pon una cota superior al número de partidos integrantes en la coalición. Puedes empezar por `NumPart =< 2`, y si los resultados son correctos, puedes ir subiendo gradualmente hasta obtener, por ejemplo, `NumPart =< 7`.
- Entre los operadores aritméticos de DES se encuentran `/\` y `\|` para realizar las operaciones de AND y OR a nivel de bit sobre dos números. Por otro lado, para calcular potencias X^Y se utiliza `X ** Y`.

Una vez que hayas definido la relación, puedes comprobar su funcionamiento mediante las siguientes consultas: ¿Qué coaliciones de 7 partidos consiguen exactamente 176 escaños? ¿Cuántas coaliciones existen que sumen exactamente 300? ¿Existe alguna con la que se obtengan exactamente 320?.