

GRADOS EN INGENIERÍA INFORMÁTICA Y EN INTELIGENCIA ARTIFICIAL
REPRESENTACIÓN DEL CONOCIMIENTO Y RAZONAMIENTO (AUTOMÁTICO)

Examen Parte I: Razonamiento Lógico. 23 de mayo de 2025

Apellidos: _____ **Nombre:** _____

- **Puntuación:** este examen de la parte I supone un tercio del total de la teoría que, a su vez, es el 50 % de la asignatura. El máximo a obtener es, por tanto, $5/3 = \mathbf{1,67}$ puntos en el total de la asignatura. La puntuación de cada apartado del examen se mide en porcentaje de ese valor.
- Utilizar preferentemente el espacio reservado en el enunciado. En caso de equivocación, solicitar un formulario nuevo.

Ejercicio 1 (20 %). Dado el siguiente programa lógico P

$$\begin{aligned} p &\leftarrow \text{not } q, \text{not } p \\ r &\leftarrow \text{not } q \\ p &\leftarrow q \\ q &\leftarrow \text{not } r \end{aligned}$$

Indica cuáles son sus modelos clásicos mediante una tabla de verdad.

p	q	r	
0	0	0	
0	0	1	
0	1	0	
0	1	1	
1	0	0	
1	0	1	×
1	1	0	×
1	1	1	×

Ejercicio 2 (30 %). Para cada uno I de los modelos clásicos obtenidos anteriormente, indica cuál es el reducto del programa anterior P^I , su modelo mínimo e indica si es o no un modelo estable. Usa tantas filas como necesites

modelo clásico I	programa reducto P^I	modelo mínimo de P^I	¿es estable? (sí/no)
$\{p, r\}$	$r \leftarrow$ $p \leftarrow q$	$\{r\}$	no
$\{p, q\}$	$p \leftarrow q$ $q \leftarrow$	$\{p, q\}$	sí
$\{p, q, r\}$	$p \leftarrow q$	$\emptyset$	no

Ejercicio 3 (10 %).

Explica qué quiere decir que un programa lógico (proposicional) sea **estratificado**. Contesta: ¿es estratificado el programa del Ejercicio 1? [] - Sí [X] - No

Un programa lógico es estratificado si no existe ningún ciclo de dependencias entre átomos que se produzca a través de la negación. Formalmente: se define un grafo de dependencias entre átomos de un programa de modo que un átomo *a* *depende* de otro *b* si *a* está en la cabeza de alguna regla donde *b* aparece en el cuerpo. La dependencia es *negativa*, si *b* aparece negado en el cuerpo. Un programa es estratificado si el grafo de dependencias no contiene ningún ciclo con al menos una dependencia negativa. El programa del Ejercicio 1 no es estratificado porque hay dos ciclos negativos: *p* depende de *not p* en la primera regla; *r* depende de *not q* en la segunda regla y, a su vez, *q* depende de *not r* en la cuarta.

Ejercicio 4 (30 %). En una base de datos del mapa de la provincia de A Coruña tenemos representados los distintos municipios de modo que el predicado **municipio(X)** indica todos los municipios que hay y el predicado **vecino(X,Y)** indica que los municipios X e Y son colindantes, es decir, tienen una frontera común. Se desea colorear el mapa usando cuatro colores, **rojo**, **verde**, **azul**, y **amarillo** de modo que no haya dos municipios vecinos pintados con el mismo color. Escribe un programa en ASP que genere todos los posibles coloreados del mapa y usa para ello el predicado **pintar(X,C)** donde X es un municipio y C el color con el que lo pintamos.

```
municipio(coruna). municipio(arteixo). municipio(cambre). ....
vecino(coruna,arteixo). vecino(coruna,culleredo). vecino(culleredo,cambre). ...
color(rojo;verde;azul;amarillo).
1 {pintar(X,C):color(C)} 1 :- municipio(X).
:- pintar(X,C), pintar(Y,C), vecino(X,Y).
#show pintar/2.
```

Ejercicio 5 (10 %). Se desea ahora obtener un coloreado del mapa de modo que el número de municipios pintados en amarillo sea el mínimo posible. Escribe el código ASP que hay que añadir al programa del ejercicio anterior que nos permita obtener la solución a este problema de optimización:

```
#minimize {1,X:pintar(X,amarillo)}.
```