

Lógica para la Computación

Convocatoria de junio. 13/01/22

Nombre:

DNI:

NOTA: Es necesario un mínimo de 3 ptos (el 50% de la puntuación total) en la prueba para sumar las prácticas correspondientes. La duración del examen es de 2 horas.

1. (1.5 ptos) Implementar un predicado PROLOG `diferencia(C1,C2,R)` que sea cierto sii `R` es el resultado de la diferencia de conjuntos $C1 \setminus C2$. Esto es, los elementos que están en `C1`, pero no en `C2`.

Ejemplo: La respuesta a la pregunta :- `diferencia([], [1,2], R)`, es `R = []`.

Ejemplo: La respuesta a la pregunta :- `diferencia([a,1,b,2,c], [1,2,3], R)`, es `R = [a, b, c]`.

Ejemplo: La respuesta a la pregunta :- `diferencia([a,1,b,2,c], [1,2,3], R)`, es `R = [a,b,c]`.

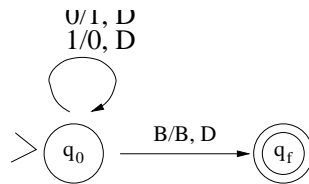
NOTA: No confundir diferencia de conjuntos con diferencia de listas.

```
diferencia([],_,[]).
```

```
diferencia([Car|Cdr],C,R) :- member(Car,C), !, diferencia(Cdr,C,R).
```

```
diferencia([Car|Cdr],C,[Car|R]) :- diferencia(Cdr,C,R).
```

2. (1.5 pts) Describir una Máquina de Turing, exclusivamente mediante un grafo de estados, que calcule el complemento a 1 de un número binario. Esto es, que sustituya en un número binario los 1's por 0's, y viceversa. Justificar la respuesta. Trazar sus movimientos para la entrada $w = 0110$.



En cuanto a la traza correspondiente a la entrada $w = 0110$, ésta sería:

$$q_0 0 1 1 0 \vdash 1 q_0 1 1 0 \vdash 1 0 q_0 1 0 \vdash 1 0 0 q_0 0 \vdash 1 0 0 1 q_0 \vdash 1 0 0 1 B q_f$$

3. (1 pto) La siguiente fórmula lógica recibe el nombre de “*axioma de inducción de la aritmética*”:

$$\forall P, [P(0) \wedge \forall X, [P(X) \Rightarrow P(X + 1)] \Rightarrow \forall Y, P(Y)]$$

que expresa que “*para cualquier conjunto P de números que verifique las propiedades:*

$$a) \quad 0 \in P$$

$$b) \quad n \in P \Rightarrow n + 1 \in P$$

entonces también verifica que $P = \mathbb{N}$ ”. Razonar si es implementable en PROLOG.

La fórmula no es implementable en PROLOG, dado que en PROLOG los funtores son siempre constantes¹, algo que aquí no ocurre con P . Dicho de otro modo, para que fuera implementable, PROLOG tendría que permitir el uso de funtores variables, cosa que no es cierta.

¹De hecho, suele definirse en ese contexto una constante como un funtor de arista nula.

4. (2 pts) Partiendo del hecho de que cualquier estrategia de construcción de un árbol de prueba da lugar a un algoritmo de resolución, indicar una ventaja y un inconveniente de la resolución SLD asociada a una construcción ~~breadth-first~~ *depth-first*. Justificar la respuesta, en cada caso, mediante un ejemplo.

Una ventaja es la gestión eficiente de la memoria, al reducir el conjunto de variables a almacenar para el tratamiento de un nodo del árbol de resolución ... a aquellas presentes en la rama actual desde la raíz de dicho árbol. Ello permite recuperar esa memoria para el tratamiento posterior de las ramas situadas más a la derecha, algo visible en cualquiera de los árboles que podéis ver en los apuntes del documento `prologIA.pdf`, accesible en el directorio

Documentos y Enlaces / Material de estudio / Prolog

de la entrada MOOVI de la asignatura.

Una desventaja es el tratamiento deficiente de la recursividad izquierda, lo que puede dar lugar a ciclos de resolución, tal y como se puede ver en la primera versión del programa que se analiza en el Ejemplo 37 del documento `prologIA.pdf`, accesible en el directorio

Documentos y Enlaces / Material de estudio / Prolog

de la entrada MOOVI de la asignatura.