EXAMEN FINAL DE LA ASIGNATURA INFORMÁTICA APLICADA

13 de SEPTIEMBRE 2000 –

TEORÍA

Tema: Espacios de representación de problemas. Búsqueda y búsqueda heurística (4 pto.)

Pregunta: Algoritmo de medios y fines (2 ptos.)

Pregunta: Mecanismos de inferencia en los sistemas basados en reglas (2 ptos.)

Pregunta: Tipos de aprendizaje (2 ptos.)

PRÁCTICAS

Practicas: .-

1.- Restricciones predicado. Restricciones que hacen uso de valores de retorno de una función. Describa ejemplos que hagan uso de estas restricciones. (2 ptos.)

2.- Dada la siguiente sintaxis de definición de módulos

```
(defmodule <nombre-del-módulo> [<commentario>]
   <especificación-portabilidad>*)

<especificación-portabilidad> ::= (export <port-item>) |
                                  (import <module-name> <port-item>)

<port-item>  ::= ?ALL |
                 ?NONE |
                 <constructor-portable> ?ALL |
                 <constructor-portable> ?NONE |
                 <constructor-portable> <nombre-constructor>+

<constructor-protable>    ::= deftemplate | defclass |
                              defglobal | deffunction |
                              defgeneric
```

Explique su significado y escriba unos ejemplos explicando el comportamiento del sistema (2 ptoS.)

3.- La representación del conocimiento del dominio en Clips. (2pto.)

4.- Explique la utilidad de cada una de las siguientes líneas de código en la solución de problema del mundo de los bloques.

```
; Regla que mueve un bloque al suelo
(defrule mover-al-suelo
        (bloque ?x)
        (bloque ?debajo)
        ?dir2 <- (movimiento-deseado ?x sobre suelo)
        (encima-de ?x nada)
        ?dir <- (encima-de ?debajo ?x)
=>
        (assert (encima-de suelo ?x))
        (assert (encima-de ?x nada))
        (assert (encima-de ?debajo nada))
        (retract ?dir)
        (retract ?dir2)
        (printout t ?x " esta en el SUELO." crlf)
)
```

5.- En el problema de las jarras de agua describa la utilidad de la siguiente regla y describa cada una líneas de código que contienen. ¿Qué hechos se están representando y cómo se representan? ¿Qué otros hechos habría que tener en cuenta en la resolución de este problema? ¿Cómo podrían representarse?

```
(defrule llena-jarra4
        (declare (salience 510))
   ?estado <- (estado
                          (jarra-agua4 ?cantidad4)
                          (jarra-agua3 ?cantidad3)
                          (profundidad ?profundidad)
        )
      (test (< ?cantidad4 4))
        (test (< ?profundidad 9)) ;
        (profundidad ?profundidad-actual)
        (test (= ?profundidad ?profundidad-actual))
=>
   (assert (estado          (jarra-agua4 4)
                            (jarra-agua3 ?cantidad3)
                            (profundidad (+ ?profundidad 1))
                            (padre ?estado)
                            (operacion "Llena la jarra de 4 l.")
     )
   )
)
```