

Apellidos:

Nombre:

Ejercicio 1 [1.5 puntos] El procedimiento (`resolventes C1 C2`) devuelve el conjunto de las resolventes entre las cláusulas `C1` y `C2`. Se considera la siguiente definición de dicho procedimiento.

```
(defun resolventes (C1 C2)
  (mapcar #'(lambda (L) (resolvente C1 C2 L))
    (intersection C1 (mapcar #'complementario C2))))
```

Explicar si la definición anterior es correcta o incorrecta y, en el caso de ser incorrecta,

1. dar un ejemplo explicando los errores de la definición y
2. decir lo que hay que añadirle a la definición para que sea correcta.

Ejercicio 2 [1.5 puntos] El procedimiento (`unifica e1 e2`) devuelve un unificador de máxima generalidad de las expresiones `e1` y `e2`, si son unificables y devuelve `fallo` en caso contrario. Se considera la siguiente definición de dicho procedimiento.

```
(defun unifica (e1 e2)
  (cond ((es-variable e1)
    (if (member e1 e2) 'fallo
      (list (cons e1 e2))))
    ((es-variable e2)
    (if (member e2 e1) 'fallo
      (list (cons e2 e1))))
    ((or (atom e1) (atom e2)) 'fallo)
    ((equal e1 e2) nil)
    (t (let ((s1 (unifica (first e1) (first e2))))
      (if (eql s1 'fallo)
        'fallo
        (let ((s2 (unifica (aplica (rest e1) s1)
          (aplica (rest e2) s1))))
          (if (eql s2 'fallo)
            'fallo
            (composicion s1 s2))))))))))
```

Explicar si la definición anterior es correcta o incorrecta y, en el caso de ser incorrecta,

1. dar ejemplos explicando los errores de la definición y
2. decir lo que hay que modificar en la definición para que sea correcta.

Ejercicio 3 [2 puntos] Se define el procedimiento

```
(defun prueba-consecuencia-por-resolucion-unidad (S F)
  (prueba-por-resolucion-unidad
    (clausulas-conjunto (append S (list (negacion F))))))
```

para determinar si la fórmula F es consecuencia del conjunto de fórmulas S utilizando la resolución unidad. Aplicando dicho procedimiento, determinar si la fórmula $\neg q \rightarrow \neg p$ es consecuencia del conjunto de fórmulas $\{p \rightarrow (q \vee r), q \leftrightarrow r\}$.

Ejercicio 4 [2 puntos] ¿Qué significa que un refinamiento de resolución sea completo?. ¿Es completa la resolución por entradas?. **Razonar** la respuesta.

Ejercicio 5 [1.5 puntos] Se considera la siguiente base de conocimiento

```
p(0,X,X) <- .
p(s(X),Y,s(Z)) <- p(X,Y,Z).
```

Construir el árbol SLD correspondiente a dicha base y al objetivo

```
<- p(s(X),X,s(s(s(0))))).
```

indicando las respuestas obtenidas.

Ejercicio 6 [1.5 puntos] Se considera la siguiente base de conocimiento en CLIPS

```
(deffacts inicio
  (numero 3)
  (numero 7)
  (numero 5))

(defrule r1
=>
  (assert (s 0) (1)))

(defrule r2
  ?h1 <- (numero ?n)
  (not (numero ?m&:(< ?m ?n)))
  ?h2 <- (s ?x)
  ?h3 <- (1 $?y)
=>
  (retract ?h1 ?h2 ?h3)
  (assert (s (+ ?x ?n))
    (1 $?y ?n)))
```

Escribir la tabla de seguimiento de su ejecución e indicar los hechos que quedan finalmente en memoria.