



UNIVERSITAT
POLITÈCNICA
DE VALÈNCIA



Escuela Técnica Superior de Ingeniería Informática



**Departamento de Sistemas Informáticos y Computación
Escuela Técnica Superior de Ingeniería Informática
Universitat Politècnica de València**

BOLETÍN DE EJERCICIOS SISTEMAS INTELIGENTES

Bloque 1: Representación del Conocimiento

Septiembre 2019

CUESTIONES

1) Dado un SBR, con una única regla:

```
(defrule regla-1
?f <- (lista ?x $?y ?x $?y ?x)
=>
  (retract ?f)
  (assert (lista $?y)))
```

, y la BH inicial {(lista a b c b c b c b c b a b c b c b c b a)}, ¿cuál será el estado final de la Base de Hechos?

- A. {(lista a)}
- B. {(lista a) (lista)}
- C. {(lista b a b)}
- D. {(lista c b c)}

2) En un determinado SBR para resolver un problema, la declaración de una BH inicial representa:

- A. Los datos estáticos del problema.
- B. Los datos dinámicos del problema, en su estado inicial.
- C. Los datos estáticos y dinámicos, en su estado inicial.
- D. Ninguna de las anteriores es cierta. La BH representa los hechos y el conjunto de reglas del dominio.

3) Supongamos que tenemos en una mesa varias cajas de distintos tamaños que queremos apilar en una torre, de mayor a menor tamaño, mediante acciones de 'apilado'. La regla meta aclara el objetivo deseado. Definimos un SBR donde la BH inicial se describe de la siguiente forma:

```
(defacts prueba (mesa 2 5 1 6 8 7 4 torre ))
```

, y se definen las siguientes reglas:

```
(defrule mesa-a-torre
(mesa $?rest1 ?x $?rest3 torre $?rest1 ?y )
(test (> ?y ?x))
=>
  (assert (mesa $?rest1 $?rest3 torre $?rest1 ?y ?x )))

(defrule mesa-a-torre-vacia
(mesa $?rest1 ?x $?rest3 torre )
=>
  (assert (mesa $?rest1 $?rest3 torre ?x )))
```

```
(defrule meta
  (mesa torre 1 2 4 5 6 7 8 )
=>
  (halt))
```

¿Cuál de las siguientes afirmaciones es correcta?

- A. El SBR definido funciona correctamente y obtendrá en la torre las cajas ordenadas por tamaño.
 - B. El SBR definido funciona correctamente, obtendrá en la torre las cajas ordenadas por tamaño, y la regla meta no es necesaria ya que el proceso acabará al quedar la mesa vacía.
 - C. El SBR solo funcionaría si las cajas en la mesa están ordenadas de menor a mayor, por ejemplo: (defacts prueba (mesa 1 2 4 5 6 7 8 torre)).
 - D. Ninguna de las afirmaciones anteriores es cierta.
-

- 4) Sea el siguiente SBR para calcular el factorial de un número, ¿cuál de las siguientes afirmaciones es correcta?

```
(defacts factorial (numero 5) (fact 1))
```

```
(defrule fact
  ?f1 <- (numero ?n1)
  ?f2 <- (fact ?n2)
=>
  (retract ?f1 ?f2)
  (assert (numero (- ?n1 1)))
  (assert (fact (* ?n2 ?n1 ))))
```

- A. El SBR funciona correctamente.
 - B. Lo anterior no es cierto, pues se requiere una condición (test (> ?n1 1)) en la premisa de la regla para que funcione correctamente.
 - C. La modificación anterior no bastaría, siendo además necesario una regla de parada: (defrule parada (declare (salience 100)) (numero 1) => (halt)).
 - D. Ninguna de las anteriores es cierta.
-

- 5) Sea el hecho:

```
(escuela clase 1 niños 15 niñas 18 clase 2 niños 21 niñas 14 clase 3 niños 16 niñas 17)
```

, donde el número que aparece después del símbolo 'clase' indica el identificador de dicha clase, y los valores numéricos que aparecen después de los símbolos 'niños' o 'niñas' indican el número de

niños o niñas de la clase correspondiente. Indica el patrón adecuado para obtener únicamente el identificador de una clase cualquiera y el número de niñas de dicha clase:

- A. (escuela \$? clase ?c \$? niñas ?na \$?) Una clase con otras niñas
 - B. (escuela \$? clase ?c niños ? niñas ?na \$?)
 - C. (escuela \$? clase ? niños \$? niñas ?na \$?) No guarda la clase
 - D. (escuela clase ?c niños ? niñas ?na) No hace pattern-matching porque le falta el último \$?
-

6) Sea un SBR formado por $BH_{inicial} = \{(lista\ 23\ 14\ 56\ 33)\}$, y las siguientes reglas:

```
(defrule R1                                (defrule R2
(declare (salience 100))                    (declare (salience 150))
?f <- (lista $?x ?z ?y $?w)                ?f <- (lista $?x ?z ?y $?w)
(test (< ?z ?y))                            (test (>= ?z ?y))
=>
(assert (lista $?x ?z ?y $?w)))            =>
                                           (assert (lista $?x ?z ?y $?w)))

(defrule final
(declare (salience 200))
(lista $?list)
=>
(halt))
```

¿Cuál sería el contenido del Conjunto Conflicto (**Agenda**) tras el primer *pattern-matching*?

- A. Una instancia de la regla R1, una instancia de la regla R2 y una instancia de la regla final
 - B. Dos instancias de la regla R2 y una instancia de la regla R1
 - C. Dos instancias de la regla R2, una instancia de la regla R1 y una instancia de la regla final
 - D. Una instancia de la regla final
-

7) Sea de nuevo el SBR de la pregunta 6. Asumiendo que la estrategia del Conjunto Conflicto (Agenda) es anchura, ¿cuál es la primera instancia de regla que selecciona el motor de inferencia de CLIPS para ser ejecutada? Indica cuál de las siguientes afirmaciones es CIERTA:

- A. Selecciona una instancia de la regla R1.
 - B. Selecciona una instancia de la regla R2.
 - C. Selecciona una instancia de la regla final.
 - D. Ninguna de las anteriores.
-

8) Sea un SBR, con una única regla:

```
(defrule R1
?f <- (lista ?x $?y ?x $?z)
=>
```

```
(retract ?f)
(assert (lista $?y ?x $?z))
(printout t "La lista se ha modificado " crlf))
```

, y la BH inicial {(lista a b a b a)}. Tras ejecutar el SBR, ¿cuántas veces se habrá mostrado en pantalla el mensaje "La lista se ha modificado"?

- A. 1
- B. 2
- C. 3
- D. 4

9) Tenemos un montacargas que recoge paquetes de la planta baja y los reparte entre diferentes plantas. Una instancia del problema tiene dos paquetes A y B, con destino 2ª y 3ª planta, peso 2 kg. y 8 kg. respectivamente. El montacargas está en la planta baja y su peso máximo es 40 kg. ¿Cuál de las siguientes representaciones NO serviría para realizar una búsqueda en grafo en una representación basada en estados?

- A. (montacargas planta 0 carga paquete A 2 2 paquete B 3 8 peso-maximo 40 nivel 0)
- B. (montacargas planta 0 carga paquete A 2 2 paquete B 3 8) (peso-maximo 40)
- C. (montacargas planta 0 carga paquete A 2 2 paquete B 3 8 nivel 0) (peso-maximo 40)
- D. (montacargas planta 0) (carga paquete A 2 2 paquete B 3 8 nivel 0) (peso-maximo 40)

10) Dado el siguiente SBR, ¿cuántas reglas se insertarán en la agenda en el primer ciclo de inferencia?

```
(defrule R1
(lista $?x1 ?y $?x2 ?y $?x3)
=>
(assert (lista $?x1 ?y $?x3)))

(deffacts inicio
(lista 2 3 1 2 3 2 1))
```

- A. 4
- B. 5
- C. Ninguna
- D. 3

11) Dado el siguiente SBR, indica cuál de las siguientes respuestas es CORRECTA:

```
(defrule R1
(declare (salience 100))
?f <- (lista $?x ?y)
```

```

      (test (> ?y 5))
=>
      (retract ?f)
      (assert (lista $?x)))

```

```

(defrule R2
  (declare (salience 200))
  ?f <- (lista ?y $?x)
  (test (> ?y 5))
=>
  (retract ?f)
  (assert (lista $?x)))

```

```

(deffacts inicio
  (lista 3 7 1 5 9))

```

- A. Sólo en el caso de que la estrategia de la agenda sea anchura, se ejecutará en primer lugar una instancia de R1
 - B. Sólo en el caso de que la estrategia de la agenda sea profundidad, se ejecutará en primer lugar una instancia de R2
 - C. Se ejecutará una instancia de R1 en primer lugar en cualquier caso
 - D. Se ejecutará una instancia de R2 en primer lugar en cualquier caso
- Aunque el salience de R2 sea de 200, no hace pattern-matching

12) Dado el siguiente hecho (pila A B A A B B A pilaA pilaB), que representa el estado inicial de un SBR, donde se tiene una pila inicial con bloques A y B y el objetivo es separar dichos bloques en dos pilas, una con bloques A y otra con bloques B. Indica cuál de las siguientes reglas NO toma un bloque A de la pila inicial y lo mueve a la pila de bloques A, de manera que se pueda resolver el problema:

- A. (defrule mover-a-pila-A
 (pila \$?x A \$?y pilaA \$?z)
 =>
 (assert (pila \$?x \$?y pilaA A \$?z)))
- B. (defrule mover-a-pila-A
 (pila \$?x ?b \$?y pilaA \$?z)
 (test (eq ?b A))
 =>
 (assert (pila \$?x \$?y pilaA A \$?z)))
- C. (defrule mover-a-pila-A
 (pila \$?x ?b \$?y pilaA \$?z)
 (test (eq ?b A))
 =>
 (assert (pila \$?x ?b \$?y pilaA ?b \$?z)))

```

D. (defrule mover-a-pila-A
    (pila $?x ?b $?y pilaA $?z)
    (test (eq ?b A))
=>
    (assert (pila $?x $?y pilaA ?b $?z)))

```

13) En un almacén se tienen dos zonas: una de carga y otra de descarga. En cada zona, puede haber varias pilas (identificadas con valores de 1 a 5) de pallets de tipo A, B o C para ser cargados o descargados en los camiones. Sea el siguiente estado inicial:

(almacen zona carga pila 1 A B C pila 2 B C B pila 3 A zona descarga pila 4 A B A pila 5 B A B B A)

Indica cuál de los siguientes patrones NO se podría utilizar para almacenar únicamente en la variable ?p el identificador de una pila de la zona de carga cuyo primer pallet sea de tipo A:

- A. (almacen zona carga \$?c pila ?p A \$?r zona descarga \$?d)
 - B. (almacen zona carga \$?c pila ?p A \$?r)
 - C. (almacen \$?c pila ?p A \$?r zona descarga \$?d)
 - D. (almacen \$?c pila ?p A \$?r 4 \$?d)
-

14) Dada la BH inicial= {(elemento e) (lista e a e b c d e f)}, y el siguiente conjunto de reglas:

```

(defrule R1
  ; (declare (salience 10))
  (elemento ?e)
  (lista $?a ?e $?b)
=>
  (assert (lista ?e $?a $?b)))

```

```

(defrule R2
  ; (declare (salience -30))
  (lista ?a $?x ?a)
  (elemento ?a )
=>
  (assert (lista $?x)))

```

¿Cuál de las siguientes afirmaciones es CORRECTA? (NOTA: el ; delante de los comandos (declare (salience ...)) indica que están comentados)

- A. El estado final dependerá de la estrategia de control aplicada (anchura, profundidad, coste uniforme, etc.)
- B. En el SBR definido no llega a lanzarse ninguna regla.
- C. Si se hubieran declarado prioridades en las reglas (salience ...), el estado final dependería de dichas prioridades
- D. El estado final será el mismo, cualquiera que sea el tipo de control que se aplique

15) Dada la siguiente parte izquierda de una regla

```
(defrule r1
  (lista $? ?x $? ?y)
  (test (< ?x ?y))
=>
...

```

, y el siguiente hecho: (lista 1 3 2 1 3 6), ¿Cuántas instancias de esta regla se incluirán en la agenda?:

- A. 0
 - B. 1
 - C. 5
 - D. Más de 5
-

16) Dada la BH={{(lista1 b a a c c a c b b c)(lista2 a c)}} y la siguiente regla

```
(defrule r1
  ?f <- (lista1 $?x ?a ?a $?y)
  (lista2 $? ?a $?)
=>
(retract ?f)
(assert (lista1 $?x ?a $?y)))

```

Indicad cuál será la Base de Hechos final.

- A. {(lista1 b b b c) (lista2 a c)}
 - B. {(lista1 b a c a c b b c) (lista2 a c)}
 - C. {(lista1 b b b c)}
 - D. {(lista1 b a c a c b c) (lista2 a c)}
-

17) Dada la siguiente parte izquierda de una regla:

```
(defrule r2
  ?f <- (lista $? ?b $?x ?b $?x)
=>
...

```

y el hecho (lista c c d c c d c c d). ¿Cuántas instancias de esta regla se insertarían en la agenda?

- A. 1
- B. 2
- C. 3
- D. 4

18) Dada la base de hechos inicial: BH={{(lista 5 7 3 1 6 4) (maximo 0)}} y la siguiente regla para calcular el máximo de una lista

```
(defrule r4
  ?f1 <- (lista $?a ?b $?c)
  ?f2 <- (maximo ?x)
  (test (> ?b ?x))
=>
  (assert (lista $?a $?c))
  (assert (maximo ?b)))
```

Si nuestro objetivo es obtener una base de hechos final (tras la ejecución sucesiva de la regla) en la cual el hecho 'maximo' solo puede aparecer una vez (conteniendo el valor máximo de la lista).

¿Cuál de las siguientes afirmaciones es CIERTA para obtener nuestro objetivo?

- A. La regla es correcta
 - B. Sería necesario añadir (retract ?f1)
 - C. Sería necesario añadir (retract ?f2)
 - D. La modificación de C no bastaría ya que sería también necesario añadir (retract ?f1)
-

19) Dado el siguiente hecho: (problema torre a b c nombre A torre a nombre B torre nombre C)
¿Cuál de los siguientes patrones serviría para obtener el nombre de una torre con un único elemento en ella?

- A. (problema \$?x torre ?a \$?y nombre ? \$z)
 - B. (problema \$?x torre ?a nombre ?z \$x)
 - C. (problema \$?x torre ?a nombre ?z \$?)
 - D. (problema \$? torre ?a nombre ?)
-

20) Sea el siguiente SBR para calcular el número de Fibonacci de un número $n > 0$, por ejemplo $n=5$ (el número de Fibonacci se calcula en fib-1), ¿cuál de las siguientes afirmaciones es correcta?

```
(defrule fact
  ?f1 <- (numero ?n1)
  ?f2 <- (fib-1 ?n2)
  ?f3 <- (fib-2 ?n3)
=>
  (retract ?f1 ?f2 ?f3)
  (assert (numero (- ?n1 1)))
  (assert (fib-1 (+?n2 ?n3)))
  (assert (fib-2 ?n2 )))
```

```
(deffacts fibonacci (numero 5) (fib-1 1) (fib-2 0))
```

NOTA: El número de Fibonacci se calcula como: $f(n)=f(n-1)+f(n-2)$ siendo $f(0)=0$ y $f(1)=1$.

- A. El SBR funciona correctamente.
- B. Lo anterior no es cierto, pues se requiere una condición (`test (> ?n1 1)`) en la premisa de la regla para que funcione correctamente.
- C. La modificación anterior no bastaría, siendo además necesario una regla de parada: (`defrule parada (declare (salience 100)) (numero 1) => (halt)`).
- D. Ninguna de las anteriores es cierta.

21) Sea el hecho:

(Mercado Calle 1 Fruta 20 Pescado 0 Calle 2 Fruta 10 Pescado 10 Calle 3 Fruta 16 Pescado 4)

donde el número que aparece después del símbolo 'Calle' indica el identificador de dicha calle en el Mercado, los números que aparecen después de los símbolos 'Fruta' o 'Pescado' indican el número de puestos de fruta o puestos de pescado en la calle correspondiente. ¿Cuál será el patrón adecuado para obtener únicamente el identificador de una calle cualquiera y el número de puestos de pescado de dicha calle?

- A. (Mercado \$? Calle ? Fruta \$? Pescado ?n \$?)
- B. (Mercado \$? Calle ?c \$? Pescado ?n \$?)
- C. (Mercado Calle ?c Fruta ? Pescado ?n)
- D. (Mercado \$? Calle ?c Fruta ? Pescado ?n \$?)

22) Sea un SBR formado por `BHinicial={{(lista 2 1 6 2 3)}}`, y las siguientes reglas:

| | |
|---|---|
| <pre>(defrule R1 ?f <- (lista \$?x ?z ?y \$?w) (test (< ?z ?y)) => (assert (lista \$?x ?z ?y \$?w)))</pre> | <pre>(defrule R2 ?f <- (lista \$?x ?z ?y \$?w) (test (> ?z ?y)) => (assert (lista \$?x ?z ?y \$?w)))</pre> |
|---|---|

¿Cuál sería el contenido del Conjunto Conflicto (Agenda) tras el primer pattern-matching?

- A. Una instancia de la regla R1 y dos de la R2
- B. Dos instancias de la regla R1 y una instancia de la regla R2
- C. Dos instancias de la regla R1 y dos instancias de la regla R2
- D. Ninguna instancia

23) Dado un SBR compuesto de la siguiente regla:

```
(defrule regla-1
  ?f <- (lista ?y $?x ?y $?x ?y)
=>
  (retract ?f)
  (assert (lista $?x)))
```

, y la BH inicial {(lista 1 2 3 2 3 2 3 2 3 2 1 2 3 2 3 2 3 2 1)}, ¿cuál será el estado final de la Base de Hechos?

- A. {(lista 3 2 3)}
- B. {(lista 1) (lista)}
- C. {(lista 2 1 2)}
- D. {(lista 1)}

24) Sea el formato de patrón (lista [nombre^s edad^s]^m) para representar el nombre y la edad de un conjunto de personas. Dada una lista determinada de personas, se quiere contar el número de ellas cuya edad está comprendida entre 18 y 65 años. Para ello se dispone del hecho que representa la lista de personas, un hecho inicial (contador 0) para contar el número de personas y la regla que se muestra a continuación. Indica la opción CORRECTA:

```
(defrule contar
  ?f1 <- (lista $?x1 ?num $?x2)
  ?f2 <- (contador ?cont)
  (test (numberp ?num)) ;; numberp devuelve TRUE si ?num es un número
  (test (and (>= ?num 18) (<= ?num 65)))
  =>
  (retract ?f2)
  (assert (contador (+ ?cont 1))))
```

- A. El SBR funciona correctamente.
- B. Para que el SBR funcione correctamente es necesario añadir solo la instrucción (assert (lista \$?x1 \$?x2)) en la RHS de la regla.
- C. Para que el SBR funcione correctamente es necesario añadir la instrucción (retract ?f1) y la instrucción (assert (lista \$?x1 \$?x2)) en la RHS de la regla.
- D. Ninguna de las anteriores

25) Dada la BH={(lista b a a a c a c b b c) (lista1 a c d e f g)} y la siguiente regla, indica cuál será la BH final.

```
(defrule R1
  ?f <- (lista $?x ?a ?a $?y)
  (lista1 $? ?a $?)
  =>
  (retract ?f)
  (assert (lista $?x ?a $?y)))
```

- A. {(lista b b b c) (lista1 a c d e f g)}
- B. {(lista b b b c)}

- C. `{{(lista b a c a c b b c) (lista1 a c d e f g)}}`
D. `{{(lista b a c b b c) (lista1 a c d e f g)}}`
-

26) Dada la base de hechos inicial: `BH={{(lista 5 7 3 1 6 4) (minimo 9999)}}` y la siguiente regla para calcular el mínimo de una lista

```
(defrule REGLA
  ?f1 <- (lista $?a ?b $?c)
  ?f2 <- (minimo ?x)
  (test (< ?b ?x))
=>
  (assert (lista $?a $?c))
  (assert (minimo ?b)))
```

Si nuestro objetivo es obtener una base de hechos final (tras la ejecución sucesiva de la regla) en la cual el hecho (minimo ...) solo puede aparecer una vez (conteniendo el valor mínimo de la lista). ¿Cuál de las siguientes afirmaciones es CIERTA para obtener nuestro objetivo?

- A. La regla es correcta
B. `Sería necesario añadir (retract ?f2)`
C. `Sería necesario añadir (retract ?f1)`
D. `Sería necesario añadir (retract ?f1) y (retract ?f2)`
-

27) Indica cuál es el resultado final CORRECTO tras ejecutar el siguiente SBR con la BH inicial=`{{(lista 34 77 34)}}`:

| | | |
|--|--|---|
| <pre>(defrule R1 (declare (salience 25)) ?f <- (lista \$?x1 ?num \$?x2) => (retract ?f) (printout t "Mensaje 1" crlf))</pre> | <pre>(defrule R2 (declare (salience 10)) ?f <- (lista ?num \$?x ?num) => (retract ?f) (printout t "Mensaje 2" crlf))</pre> | <pre>(defrule R3 => (printout t "Mensaje 3" crlf))</pre> |
|--|--|---|

- A. Mostrará tres veces el "Mensaje 1".
B. Mostrará una vez "Mensaje 1" y una vez "Mensaje 2".
C. `Mostrará una vez "Mensaje 1" y una vez "Mensaje 3".`
D. Mostrará una vez "Mensaje 1", una vez "Mensaje 2" y una vez "Mensaje 3".
-

28) El siguiente hecho representa un conjunto de pilas y los bloques que contiene cada una de ellas. El número que aparece después del símbolo 'pila' es el identificador de la pila y a continuación aparecen los bloques de la pila, siendo el primer bloque el tope de la pila. Indica cuál sería la LHS de una regla para que se instancie CORRECTAMENTE el bloque que se encuentra en la base de cualquier pila del hecho que se muestra a continuación:

(problema pila 1 A F G J K pila 2 B D pila 3 C H I L pila 4)

- A. (problema \$?x1 pila ?num \$?x2 ?y pila \$?x3) (test (not (member pila \$?x1)))
- B. (problema \$?x1 pila ?num \$?x2 ?y pila \$?x3) (test (not (member pila \$?x2)))
- C. (problema \$?x1 pila ?num \$?x2 ?y \$?x3)(test (not (member pila \$?x3)))
- D. Ninguna de las anteriores.

29) Sea un SBR formado por BHinicial={{(lista 2 1 5 3)}}, y las siguientes reglas:

| | |
|--|---|
| <pre>(defrule R1 (declare (salience 100)) ?f <- (lista \$?x ?z ?y \$?w) (test (< ?z ?y)) => (assert (lista \$?x ?z ?y \$?w)))</pre> | <pre>(defrule R2 (declare (salience 150)) ?f <- (lista \$?x ?z ?y \$?w) (test (>= ?z ?y)) => (assert (lista \$?x ?z ?y \$?w)))</pre> |
|--|---|


```
(defrule final
  (declare (salience 200))
  (lista $?list)
=>
  (halt))
```

tras el primer *pattern-matching*, ¿cómo quedarían ordenadas las instancias el Conjunto Conflicto (Agenda)?

- A. Una instancia de la regla R1, una instancia de la regla R2 y una instancia de la regla final
- B. Dos instancias de la regla R2, una instancia de la regla R1, una instancia de la regla final
- C. Una instancia de la regla final, dos instancias de la regla R2, una instancia de la regla R1.
- D. Una instancia de la regla final

30) Sea un SBR formado por BHinicial={{(lista 2 1 6 2 3) (elemento 5)}}, y la regla que se muestra a continuación. ¿Cuál sería el contenido final de la BH?

```
(defrule REGLA
  ?f <- (lista $?x ?z $?w)
  (elemento ?y)
  (test (< ?z ?y))
=>
  (assert (lista $?x $?w))
  (retract ?f))
```

- A. {{(lista 6) (elemento 5)}}
- B. {{(lista 2 1 2 3) (elemento 5)}}
- C. {{(lista) (elemento 5)}}
- D. {{(lista 2 2) (elemento 5)}}

31) Sea la siguiente regla para calcular el Máximo Común Divisor (mcd) de dos números enteros positivos. Indica la respuesta **CORRECTA**:

```
(defrule mcd
  ?a <- (num ?n1)
  ?b <- (num ?n2)
  (test (> ?n1 ?n2))
=>
  (retract ?a)
  (assert (num (- ?n1 ?n2))))
```

- A. **Calcula correctamente el mcd quedando un hecho 'num' con dicho valor**
 - B. Es necesario añadir una regla parada sin prioridad para que el sistema no entre en una ejecución sin fin
 - C. Es necesario añadir una regla parada con prioridad para que el sistema no entre en una ejecución sin fin
 - D. Ninguna de las anteriores es correcta
-

32) Dado un SBR cuya Base de Hechos inicial es (lista b a c c a b b a resto), y una única regla:

```
(defrule pasar
  ?a <- (lista $?x ?y ?y $?x $?z resto $?m)
=>
  (retract ?a)
  (assert (lista $?x $?x $?z resto $?m ?y)))
```

El contenido final de la Base de Hechos será:

NO HAY PATTERN-MATCHING

- A. (lista b a a b b a resto c)
 - B. (lista b a resto c a b)
 - C. (lista b a b a resto c b)
 - D. **Ninguna de las anteriores**
-

33) Dado el siguiente hecho: (Dueños coches a b c dueño P coches d dueño Q coches e f dueño R), donde se relaciona los coches y posteriormente su dueño ¿Cuál de los siguientes patrones serviría para obtener el nombre del dueño de un solo coche?

- A. (Dueños \$?x coches ?a dueño \$?z)
 - B. **(Dueños \$? coches ? dueño ?z \$?)**
 - C. (Dueños \$?x coches ?a dueño ?z \$?x)
 - D. (Dueños \$? coches ? dueño ?z)
-

34) Sea un SBR, con la BH inicial {(lista A B C A B C C B A C B A)}, y la siguiente regla:

```

(defrule regla1
  ?f1 <- (lista $?x1 ?y $?x2 ?y $?x3)
  (test (> (length $?x2) 0))
  (test (not (member ?y $?x2)))
=>
  (retract ?f1)
  (assert (lista $?x1 ?y ?y $?x3)))

```

Si se ejecuta este SBR, el resultado es:

En anchura va desde el final,
En profundidad va desde el principio,
PERO en AMBOS CASOS DE LA MISMO.

- A. Una lista que solo contiene letras A
- B. Una lista que solo contiene letras B
- C. Una lista que solo contiene letras C
- D. Dependerá de la estrategia de control que se aplique, sea anchura o profundidad.

35) Dado el hecho (prueba 1 2 3 4 5 6 7 8 9 10) y la regla :

```

(defrule regla1
  ?f1 <- (prueba $?a $?c)
=>
  (retract ?f1)
  (assert (lista $?c)))

```

Primero el vacío, y después todos los elementos

En el primer ciclo inferencial:

-> 1 + 10 = 11

- A. No se producirá ninguna instanciación
- B. Se producirán 9 instanciaciones
- C. Se producirán 10 instanciaciones
- D. Se producirán 11 instanciaciones

36) Dada la base de hechos inicial: BH={{(lista 3 5 2 5 3 4 2 9 8 8 9 6) (numero 5) (repeticiones 0)}} y la siguiente regla para calcular el número de repeticiones de un elemento de una lista de números naturales

```

(defrule REGLA
  ?f1 <- (lista $?a ?b $?c)
  ?f2 <- (numero ?x)
  ?f3 <- (repeticiones ?z)
  (test (= ?b ?x))
=>
  (assert (lista $?a $?c))
  (assert (repeticiones (+ 1 ?z))))

```


Si nuestro objetivo es obtener una base de hechos final (tras la ejecución sucesiva de la regla) en la cual el hecho (repeticiones ...) solo puede aparecer una vez (conteniendo el número de repeticiones del número indicado en el patrón (numero ..) en la lista). ¿Cuál de las siguientes afirmaciones es CIERTA para obtener nuestro objetivo?

- A. La regla es correcta
- B. Sería necesario añadir (retract ?f1)
- C. Sería necesario añadir (retract ?f1) y (retract ?f3)
- D. Sería necesario añadir (retract ?f3)

37) Sea un SBR formado por BInicial=({(lista 2 1 5 3)}), y las siguientes reglas:

| | |
|--|--|
| <pre>(defrule R1 (declare (salience 200)) ?f <- (lista \$?x ?z ?y \$?w) (test (< ?z ?y)) => (assert (lista \$?x ?z ?y \$?w)))</pre> | <pre>(defrule R2 (declare (salience 50)) ?f <- (lista \$?x ?z ?y \$?w) (test (>= ?z ?y)) => (assert (lista \$?x ?z ?y \$?w)))</pre> |
| <pre>(defrule final (declare (salience 150)) (lista \$?list) => (halt))</pre> | |

tras el primer *pattern-matching*, ¿como quedarían ordenadas las instancias el Conjunto Conflicto (Agenda)?

- A. Una instancia de la regla R1, una instancia de la regla final, y dos instancias de la regla R2
- B. Dos instancias de la regla R2, una instancia de la regla R1, una instancia de la regla final
- C. Una instancia de la regla final, dos instancias de la regla R2, una instancia de la regla R1
- D. Una instancia de la regla final

38) Dada la siguiente parte izquierda de una regla

```
(defrule r1
  (lista $?x $?w ?y ?z $?x)
=>
  ....
```

, y el siguiente hecho: (lista a b a b c c), ¿cuántas instancias de esta regla se incluirán en la agenda?:

- A. 0
- B. 1
- C. 3
- D. 5