

Sistemes Intel·ligents – Test Bloc 1 (tipus B)
ETSINF, Universitat Politècnica de València
30 octubre 2017

Cognoms:

Nom:

Grup: A B C D E F Flip

Marca solament una resposta entre les opcions donades. Cada encert sumarà 1 punt i cada error descomptarà 1/3 punts (les qüestions no contestades no afectaran a la nota)

- 1) Donada la base de fets inicial: $BF = \{(llista\ 5\ 7\ 3\ 1\ 6\ 4)\ (minim\ 9999)\}$ i la següent regla per a calcular el mínim d'una llista

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

El nostre objectiu és obtenir una base de fets final (després de l'execució successiva de la regla) en la qual el fet (minim ...) solament pot aparèixer una vegada (contenint el valor mínim de la llista). Quina de les següents afirmacions és CERTA per a obtenir el nostre objectiu?

- A. La regla és correcta
- B. Seria necessari afegir (retract ?f1)
- C. Seria necessari afegir (retract ?f2)
- D. Seria necessari afegir (retract ?f1) i (retract ?f2)

-
- 2) Indica quin és el resultat final CORRECTE després d'executar el següent SBR amb la BF inicial = {(llista 34 77 34)}:

```
(defule R1
(declare (salience 25))
?f <- (llista $?x1 ?num $?x2)
=>
(retract ?f)
(printout t "Missatge 1" crlf))
```

```
(defrule R2
(declare (salience 10))
?f <- (llista ?num $?x ?num)
=>
(retract ?f)
(printout t "Missatge 2" crlf))
```

```
(defrule R3
=>
(printout t "Missatge 3" crlf))
```

- A. Mostrarà tres vegades el "Missatge 1".
- B. Mostrarà una vegada "Missatge 1" i una vegada "Missatge 2".
- C. Mostrarà una vegada "Missatge 1", una vegada "Missatge 2" i una vegada "Missatge 3".
- D. Mostrarà una vegada "Missatge 1" i una vegada "Missatge 3".

- 3) Siga un SBR format per BFinicial={{(llista 2 1 6 2 3) (element 5)}} i la regla que es mostra a continuació. Quin seria el contingut final de la BF?

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

- A. {(llista 2 1 2 3) (element 5)}
- B. {(llista 2 2) (element 5)}
- C. {(llista 6) (element 5)}
- D. {(llista) (element 5)}

- 4) Donada la BF={{(llista b a a a c a c b b c) (llista1 a c d e f g)}} i la següent regla, indica quina serà la BF final.

```
(defrule R1
  ?f <- (llista $?x ?a ?a $?y)
    (llista1 $? ?a $? )
=>
  (retract ?f)
  (assert (llista $?x ?a $?y)))
```

- A. {(llista b a c a c b b c) (llista1 a c d e f g)}
- B. {(llista b b b c) (llista1 a c d e f g)}
- C. {(llista b b b c)}
- D. {(llista b a c b b c) (llista1 a c d e f g)}

- 5) Siga un SBR format per BFinicial={{(llista 2 1 5 3)}} i les següents regles:

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

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

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

després del primer *pattern-matching*, com quedarien ordenades les instàncies del Conjunt Conflicte (Agenda)?

- A. Una instància de la regla final.
- B. Una instància de la regla R1, una instància de la regla R2 i una instància de la regla final.
- C. Dues instàncies de la regla R2, una instància de la regla R1, una instància de la regla final.
- D. Una instància de la regla final, dues instàncies de la regla R2, una instància de la regla R1.

- 6) El següent fet representa un conjunt de piles i els blocs que conté cadascuna d'elles. El nombre que apareix després del símbol 'pila' és l'identificador de la pila i a continuació apareixen els blocs de la pila, sent el primer bloc el topall de la pila. Indica quina seria la LHS d'una regla perquè s'instancie CORRECTAMENT el bloc que es troba en la base de qualsevol pila del fet que es mostra a continuació:

(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 \$?x2)))
 - B. (problema \$?x1 pila ?num \$?x2 ?y pila \$?x3) (test (not (member pila \$?x1)))
 - C. (problema \$?x1 pila ?num \$?x2 ?y \$?x3) (test (not (member pila \$?x3)))
 - D. Cap de les anteriors.
-

- 7) Siga el format de patró (llista [nom^s edat^s]^m) per a representar el nom i l'edat d'un conjunt de persones. Donada una llista determinada de persones, es vol comptar el nombre d'elles amb edat compresa entre 18 i 65 anys. Per a açò es disposa del fet que representa la llista de persones, un fet inicial (comptador 0) per a comptar el nombre de persones i la regla que es mostra a continuació. Indica l'opció CORRECTA:

(defrule comptar

?f1 <- (llista \$?x1 ?n \$?x2)

?f2 <- (comptador ?compte)

(test (numberp ?n))

;; numberp retorna TRUE si ?n és un nombre

(test (and (>= ?n 18)(<= ?n 65)))

=>

(retract ?f2)

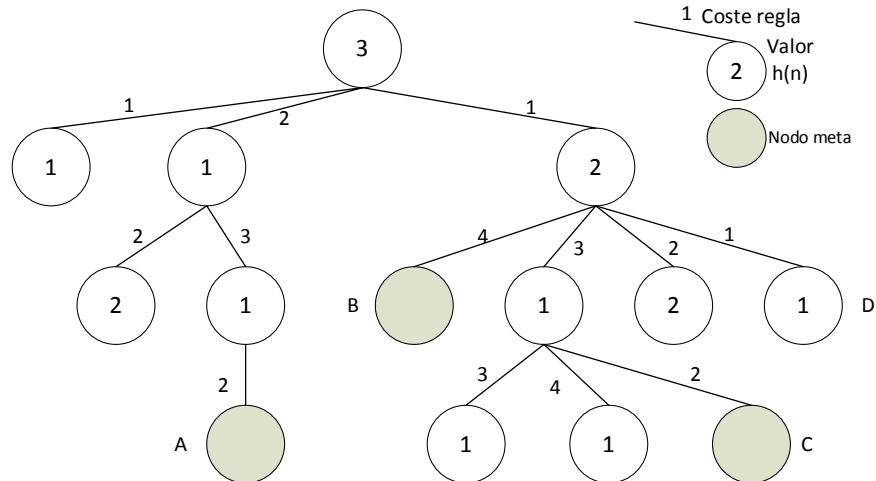
(assert (comptador (+ ?compte 1))))

- A. L'SBR funciona correctament.
 - B. Perquè el SBR funcione correctament és necessari afegir la instrucció (retract ?f1) i la instrucció (assert (llista \$?x1 \$?x2)) en la RHS de la regla.
 - C. Perquè l'SBR funcione correctament és necessari afegir solament la instrucció (assert (llista \$?x1 \$?x2)) en la RHS de la regla.
 - D. Cap de les anteriors.
-

- 8) Siga un problema de cerca en el qual els costos dels operadors són diferents. L'aplicació d'un algorisme GRAPH-SEARCH de Cost Uniforme amb control de nodes repetits retorna una solució de cost 'c' en un nivell de profunditat 'd'. Indica quina de les següents afirmacions és CERTA.

- A. Un algorisme d'Aprofundiment Iteratiu sobre el mateix problema retornarà sempre la solució òptima.
- B. No és necessari especificar un nivell de profunditat màxima per a l'espai de cerca amb la finalitat d'evitar que l'algorisme entre en un bucle infinit.
- C. Un algorisme d'Amplària sobre el mateix problema retornarà sempre la solució òptima.
- D. Una solució amb cost c' tal que c' > c solament es trobarà en un nivell d' tal que d' > d.

9) Si s'aplica un algorisme de tipus A en l'espai d'estats de la figura següent, quin node meta es triarà en primer lloc com a solució?

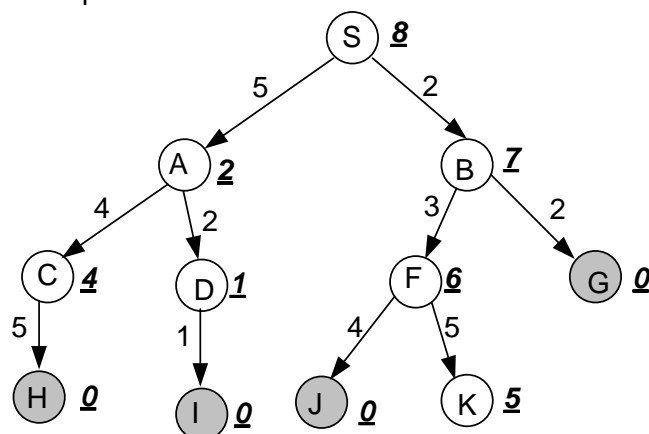


- A. A
- B. B
- C. C
- D. D

10) Donat l'espai de cerca de la figura anterior, una cerca A que utilitze dits valors, seria:

- A. Ni admissible ni consistent
- B. Admissible i consistent
- C. Admissible i no consistent
- D. No admissible però si consistent

11) Donat el següent espai de cerca, on s'indica el cost en les branques i l'estimació $h(n)$ a la dreta de cada node, indica la resposta CORRECTA.



- A. Un algorisme A generarà menys nodes que una cerca de cost uniforme.
- B. Una expansió en amplària, començant per l'esquerra, generarà igual o menor nombre de nodes que cost uniforme.
- C. Un algorisme A obtindrà la senda òptima.
- D. Cap de les anteriors és certa.

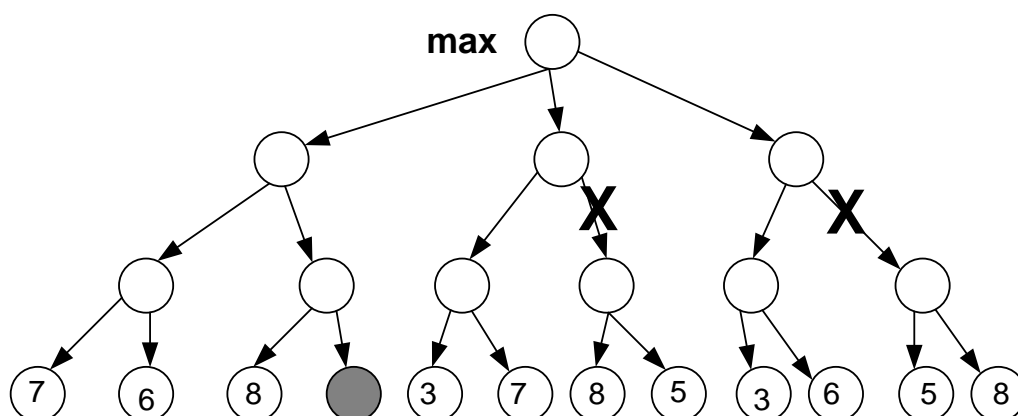
12) En l'aplicació d'una cerca A amb mètode GRAPH-SEARCH, s'ha trobat el camí òptim fins a cada node expandit si:

- A. La funció heurística és admissible.
 - B. La funció heurística és consistent.
 - C. S'efectua un control de nodes repetits en la llista OPEN.
 - D. Cap de les anteriors.
-
-

13) Siga un algorisme A amb una heurística $h(n)$. El nombre de nodes generats per a l'obtenció de la solució davant un estat inicial concret (indicar la resposta CORRECTA):

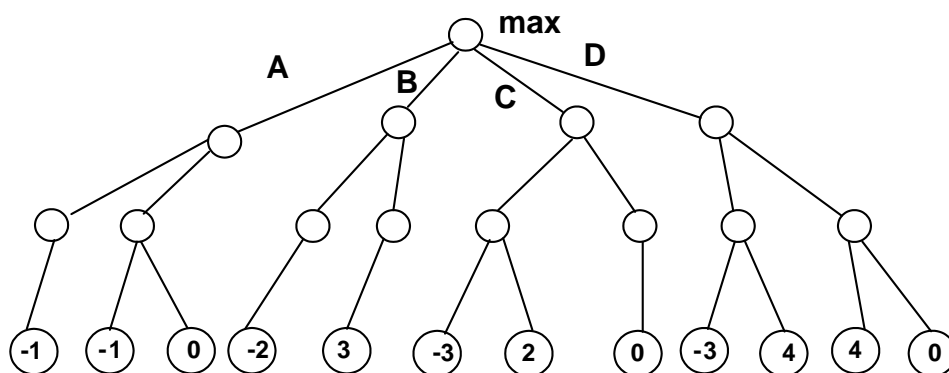
- A. Dependrà del factor efectiu de brancatge de l'heurística $h(n)$ i de la profunditat de la solució òptima.
 - B. Si $h(n)$ és admissible, mai serà major que el nombre de nodes generats amb una heurística $h'(n)$ tal que $h'(n) = h^*(n)$.
 - C. Dependrà del cost d'aplicació de cada regla.
 - D. Cap de les anteriors.
-
-

14) Donat l'arbre de joc de la figura i aplicant un procediment alfa-beta, quin valor hauria de tenir el NODE terminal ombrejat perquè es produïsquen els talls indicats en la figura?



- A. Amb qualsevol valor es produïrien els dits talls
- B. Major o igual a 7
- C. Major o igual a 8
- D. Menor o igual a 7

15) Donat l'arbre de joc de la figura, Quants nodes evitem generar respecte a un algorisme MINIMAX si realitzem una exploració alfa-beta?



- A. 6
- B. 5
- C. 4
- D. 3