**PRÁCTICA 2.1 CLIPS 10**

**INTELIXENCIA ARTIFICIAL**

Alex Baquero Domínguez

*alex.baquero@rai.usc.es*

Cuestiones 1.1:

**• ¿Qué devuelve Clips al añadir un hecho a la Base de Hechos (BH)?**

A través do comando assert (), CLIPS devolve unha mensaxe do estilo <Fact-n> (n é o numero de “hecho” según o orden no que se insertou no CLIPS) e asigna un identificador de “hecho”, empezando desde f-0 .

Ao poner reset, este “hecho” inicial é por defecto f-0

**• ¿Qué ocurre cuando se introduce un hecho “repetido” en la base de hechos (apellido-1 Perez)?**

CLIPS devolve unha mensaxe ao estilo <Fact-n> pero non introduce este dato novo na base de “hechos”. De esta maneira, n é o número que se lle asignou ao primeiro “hecho” que foi introducido antes de poner o repetido.

Cuestiones 1.2:

**¿Se ha activado alguna regla? ¿Qué hechos activan cada regla?**

A primeira activouse cando o nome indicado foi introducido na base de “hechos” mentras que a segunda cando o nome e dous apelidos indicados foron introducido na base de “hechos.

Podese ver que as regras activanse cando os antecedentes estén na base de “hechos”. As regras executaranse cando os datos metenses a través dos assert. Se xa están dentro antes de realizar o defrule, as regras activanse ao executar este.

Cuestiones 1.3:

**• ¿Qué regla se ejecuta en primer lugar? ¿Por qué?**

O orden dalo o sistema CLIPS ainda que se pode establecer unha orde de prioridade a través do salience. A regra que primeiro se executou foi a regra 2, pois a medida que se van activando introducense na pila e executanse unha a unha a medida que se extraen.

**• ¿Qué pasa si reiniciamos con (clear)?**

Cando poñemos (clear) todo o que está almacenada na base de “hechos” borrase: hechos e regras.

Cuestiones 1.4:

**• Si se introducen los hechos con (deffacts), y se carga el fichero ¿qué ocurre en la BH y en la Agenda?**

Facendo un load simplemente sen executar reset, no pasa nada na BH e Agenda. Só se executa por mensaxe:

$\*\*

TRUE

**• ¿Qué ocurre en la BH y en la Agenda al ejecutar (reset)?**

Os “hechos” activanse na BH e activanse as regras correspondientes. Pasa o mesmo que introducindo o datos por mensaxe pero introducense todos a vez.

**• ¿Cuál es el primer hecho que se ha almacenado en la BH?**

Almacenouse no orde que se puxo no .clp.

Texto

Descripción generada automáticamente

Neste caso, o nome Juanjo.

Cuestiones 1.5

**¿Qué devuelve el programa cuando el agente es un coche y el semáforo para los coches está en rojo? Copia y pega el resultado mostrado en la Dialog Window tras ejecutar el programa.**

**Texto

Descripción generada automáticamente**

Cuestiones 2.3

**Utilizando el sistema, indicar la respuesta a las cuestiones siguientes:**

Definemos os “hechos” a modo que unha persoa é pai ou nai de outra persoa ou unah persoa é home ou muller. Xeramos unha regra para elixir entre ás tres preguntas para activar unhas regras o otras.

**1. ¿Quiénes son los hermanos, tíos, sobrinos, primos y abuelos de Archie?**

Texto

Descripción generada automáticamente

**2. ¿Quiénes son abuelos y de quién?**

**Texto

Descripción generada automáticamente**

**3. ¿Quiénes son tíos y de quién?**

**Texto

Descripción generada automáticamente**

Cuestiones 2.4

**Incluir los hechos y, de ser necesario, las nuevas reglas que tengan en cuenta la nueva situación e indicar cuáles son ahora las respuestas a las cuestiones anteriores.**

**Texto

Descripción generada automáticamente**

**Texto

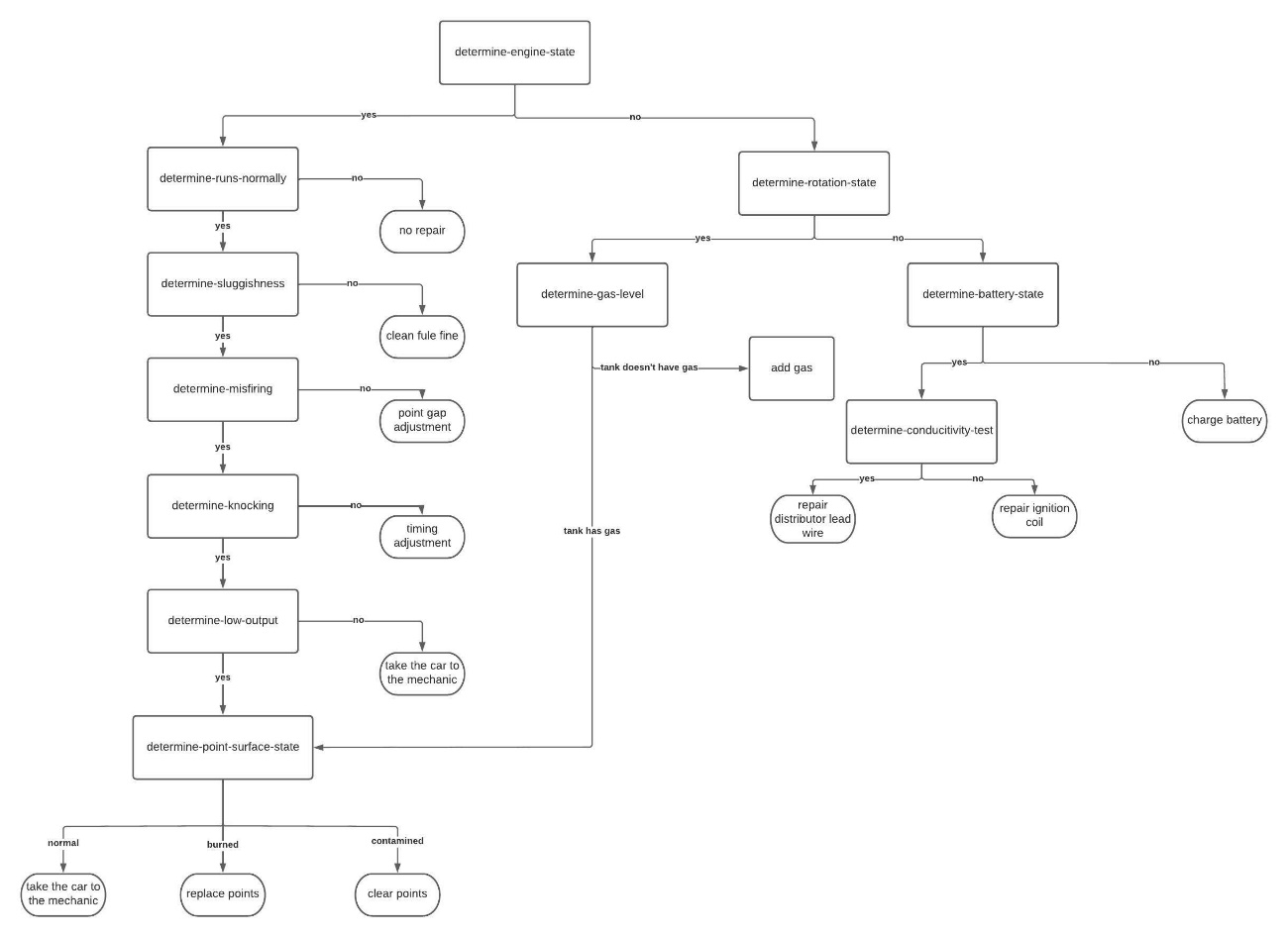
Descripción generada automáticamente**

**Texto

Descripción generada automáticamente**

Cuestiones 3.

**Analiza el orden de ejecución de las reglas del bloque QUERY RULES y represéntalo gráficamente en forma de árbol, donde los nodos son las diferentes reglas (determine-engine-state, determine-runs-normally, …)**

****

**Justifique por qué todas las reglas del bloque QUERY RULES incluyen el antecedente (not (repair ?)) y si este es necesario o podría prescindirse de él.**

Si se introduce un “hecho” como por exemplo, engine-rotates, activanse varias regras a vez. Estas vanse executando ata que o programa remata cando respondese YES. Para decidir isto, e que remate, insertase un “hecho” repair ao activar un repair rules facendo que a compobracion not(repair ¿) devolve un boolean que é falso.

En conclusión, o antecedente not repatir non se pode prescindir de él pois é necesario para saber se hai que seguir executando as regras do CLIPS cando se introduce un “hecho” . De esta manera, podemos ver se o problema esta solucionado ou non.

* Si é YES, o CLIPS non debe seguir pregutando ao usuario porque indica que o problema xa ten unha solución considerada.
* Si é NO, é necesario considerar o antecedente.

**Identifica qué reglas del sistema utilizan la variable saliencia. Revisando la documentación de CLIPS y otras fuentes investiga acerca del concepto de saliencia y justifica el papel que juega dicha variable y los valores elegidos en las reglas del sistema.**

A variable saliencia utilizase en tres regras do sistema:

* Regra system-banner: Usase como un impresor ao principio do programa. Daselle un valor igual a 10 para que sexa uns dos primeiros en executarse ou o primeiro.
* Regra print-repair. Tras coñecer o motivo da avería, esta regra activase mostrando unha mensaxe de información acerca do que se debe reparar. Daselle un valor igual a 10 para que sexa uns dos primeiros en executarse.
* Regra repairs: mostra unha mensaxe de que o usuario ten que ir a un taller para que sexa observado por un mecánico. Daselle un valor igual a -10 para que sexa sempre a última en executarse.

De esta maneira, dentro dunha regra, o concepto saliencia é a prioridade que se lle da a ese regra para executarse con respecto a outras. Por lo tanto, as regras que teñan unha variable saliencia máis alta, executaranse primeiro (de -10000 a 10000, con valor por defecto igual a 0). A evaluación de esta variable realizase:

* Tras cada ciclo de ejecución.
* Cando se activa a regra ou cando se define.