



Práctica de Inteligencia Artificial. Sistemas de Producción en CLIPS Implementación de un Juego de Simulación Social Curso 2009-10 (Septiembre)

1. Introducción

La práctica de septiembre es parecida a la propuesta en febrero con algunas modificaciones. Se ha optado por redactarla completamente en vez de destacar las modificaciones solamente. Las modificaciones principales tienen que ver con la forma de incrementar los indicadores y el estado emocional; las modalidades de juego se sustituyen por una única en que se establece un sistema de puntuación de cada jugada, y las pruebas para validar el trabajo que serán exclusivamente individuales.

2. Descripción

El objetivo de esta práctica es la implementación en CLIPS de un sistema de producción para simular un juego de sociedad virtual tipo los SIMS.

El juego consiste en la simulación del comportamiento de dos o más personajes virtuales que actúan de forma autónoma en un mismo escenario. En cada turno, uno de los personajes elegirá y ejecutará una sola acción de entre todas las posibles acciones que puede realizar y pasará el turno al siguiente personaje. Así indefinidamente hasta llegar al final del juego.

Los personajes vienen caracterizados por unos *indicadores* y un estado emocional. Los *indicadores* representan las necesidades básicas y pueden tomar valores entre 0, cuando la necesidad está completamente satisfecha, y 1000 en caso contrario. Se consideran los siguientes *indicadores*: hambre, sed, suciedad, cansancio y aburrimiento.

Para poder rebajar cada uno de los *indicadores* el personaje puede realizar las siguientes acciones:

- Comer: decrementa el *indicador* de hambre en 10 unidades.
- Beber: decrementa el *indicador* de sed en 10 unidades.
- Lavarse: decrementa el *indicador* de suciedad en 10 unidades.
- Descansar: decrementa el *indicador* de cansancio en 10 unidades.
- Jugar: decrementa el *indicador* de aburrimiento en 10 unidades.

Cada acción de las anteriores, además de rebajar en diez unidades el *indicador* correspondiente, incrementa en 1 el resto de *indicadores*.

Para poder realizar las acciones anteriores el personaje requiere tener el objeto adecuado, por ejemplo, para comer debe tener comida, para descansar debe estar en una cama o un sofá, para lavarse necesita una ducha, una bañera o un lavabo, etc. Se supone que cada objeto está en una posición y el personaje tendrá que estar situado en el mismo lugar que el objeto para poder realizar la acción (puede haber varios objetos en la misma posición).

Los personajes tendrán distintas preferencias por cada uno de los objetos, que influirán en el estado emocional del personaje. Las preferencias toman valores enteros entre 0 y 10, 0 representa que el personaje detesta

dicho objeto y 10 que le gusta mucho. El estado emocional se representa como el grado de infelicidad y toma valores entre 0 y 1000, 0 representa el mayor grado de felicidad y cuanto mayor sea su valor mayor será la infelicidad del personaje. Cuando un personaje realiza una acción con un objeto cuya preferencia vale p se incrementa su infelicidad en $(10 - p)$.

Además de esas acciones, los personajes pueden realizar las siguientes acciones:

- Trabajar: para trabajar el personaje tiene que estar situado junto a un ordenador (tele-trabajo). El trabajo incrementa la riqueza del personaje en 10 unidades e incrementa todos los *indicadores* en 1.
- Comprar: para comprar el personaje tiene que estar situado junto al ordenador (compra por internet). Los únicos objetos que necesita comprar son la comida y la bebida puesto que el resto de objetos asumimos que están en el escenario inicialmente y no se deterioran. La comida y bebida que compran debe situarse en la nevera o despensa de la cocina y sólo puede ser consumida por el personaje que la ha comprado. Decrementa en 1 unidad la riqueza por cada objeto que compra e incrementa todos los *indicadores* en 1.
- Moverse: con esta acción los personajes pueden moverse de la posición en la que se encuentran a otra cualquiera dentro del escenario. Incrementa todos los *indicadores* en 1 y la infelicidad en 5.

Se puede asumir que en el escenario hay dos ordenadores, uno para trabajar y otro para comprar, con preferencias distintas, o bien, que sólo hay un ordenador en cuyo caso el estado emocional de los personajes varía de forma similar tanto al trabajar como al comprar.

Cada vez que los personajes realizan una acción de comer o beber el objeto que utilizan desaparece del escenario. Con el resto de acciones el objeto involucrado permanece igual. Todos los objetos que hay en el escenario pueden ser usados por cualquier personaje, excepto la comida y bebida que sólo la puede consumir el personaje que la ha comprado.

El objetivo del juego es conseguir la máxima puntuación posible. Inicialmente todos los personajes empiezan con 0 puntos. Cada vez que un personaje realiza una acción se incrementa su puntuación de la siguiente manera:

$$\Delta puntuacion = (1000 - infelicidad) + \sum_{indicador} (1000 - indicador)$$

Por ejemplo, si en una jugada un personaje que tiene todos los indicadores a 100 y un valor de la infelicidad igual a 0, realiza una acción de beber con una bebida cuya preferencia es de 8, tras la ejecución de dicha acción el indicador de sed valdrá 90 y el resto de indicadores 101 y la infelicidad se incrementará en 2 unidades $(10-8)$. Por tanto, la puntuación de dicho personaje se incrementará en $(1000 - 2) + (1000 - 101) \times 4 + (1000 - 90) = 5504$.

En cada turno se incrementa el tiempo total del juego en una unidad. Cuando se agote el tiempo establecido al principio del juego ganará el personaje que tenga mayor puntuación.

3. Se pide

1. Construir la ontología, con las clases y plantillas necesarias para modelar el conocimiento del sistema en el fichero **ontologia.clp**.
2. Definir las reglas, las instancias y hechos necesarios para poder realizar el control del juego en el fichero **servidor.clp**. Las reglas que se deben incluir son aquellas que permitan establecer los distintos turnos entre los jugadores y las que determinan cuándo finaliza el juego y quién es el ganador. Los valores de las variaciones de los *indicadores*, estado emocional y riqueza deben representarse en el servidor de forma que puedan cambiarse fácilmente.
3. Definir las reglas, instancias y hechos necesarios para simular el comportamiento de dos personajes virtuales con el mismo comportamiento, en el fichero **cliente.clp**. La estrategia de control debe ser *random* y habrá que modelar mediante reglas el comportamiento de los personajes para que intenten ganar el juego. Por medio de reglas habrá que implementar el tipo de acción que más le conviene realizar al personaje,

según el valor de los *indicadores*. Una vez que se establece el tipo de acción, habrá que diseñar las reglas para que utilicen el objeto que más felicidad les proporciona.

4. Simular 5 partidas, como mínimo, en que juegan dos o más personajes variando las preferencias por los distintos objetos y el nº de objetos iniciales en el escenario.

Para cada jugada definir el fichero **jugada1.clp**, **jugada2.clp**, ... con dichos valores. La traza de las partidas realizadas hay que entregarla y en la memoria hacer un pequeño resumen de las pruebas y resultados obtenidos. Con el comando *dribble-on "fichero"* se consigue que todos los *printout* del programa se salven en un fichero.

4. Recomendaciones

- No se pueden usar **ifs** ni cualquier otra sentencia de programación estructurada en las reglas.
- No se deben usar **tests** muy cargados en la parte izquierda de las reglas.
- Se debe evitar el uso de funciones y variables globales. En caso de estimar necesario su uso consultad con alguno de los profesores de prácticas.
- Procurad diseñar el sistema basado en reglas de forma que las reglas sean escuetas con un objetivo sencillo y claro, y fáciles de comprender, evitando al mismo tiempo la proliferación de reglas demasiado simples. Se deben utilizar plantillas que representen conocimiento, de forma que las reglas sean lo más generales posibles
- El agrupamiento de varias reglas en una sola utilizando OR no significa que la regla obtenida esté simplificada. Es preferible construir varias reglas porque facilita la comprensión.
- Evitad los test de igualdad que comparan dos variables. No son necesarios, basta con poner el mismo nombre de variable.
- El código deberá ser lo más legible posible, incluyendo la indentación y los comentarios adecuados.
- La documentación sobre CLIPS se puede encontrar en:
<http://scalab.uc3m.es/docweb/ia/transparencias.html>
- El manual de referencia de CLIPS, en html:
http://scalab.uc3m.es/docweb/ia/CLIPS_REFERENCIA
- Las diferentes versiones de CLIPS se pueden encontrar en:
<http://scalab.uc3m.es/docweb/ia/software.html>

5. Entrega de la práctica

La fecha límite para entregar la práctica será el día del examen de septiembre. Hay que entregar una memoria en papel y los ficheros correspondientes a la práctica cuya entrega se realizará por Aula Global. **Es muy importante seguir estrictamente las normas de entrega aquí descritas.**

La memoria se entregará directamente a los profesores de prácticas o se dejará en uno de sus casilleros.

- La memoria debe tener la siguiente estructura:
 1. Portada con los nombres y direcciones de correo de los autores.
 2. Introducción. Una hoja de descripción de la práctica desde vuestro punto de vista, no desde la copia del enunciado.
 3. **Código y explicación** de:

- Estructura estática de la memoria de trabajo (plantillas y jerarquía de marcos).
 - Las reglas que representan las inferencias del sistema de producción. Se debe incluir una explicación general sobre cómo están organizadas estas reglas (por ejemplo, si hay varias reglas que realizan una determinada función).
4. Experimentación: se debe incluir la explicación de los experimentos realizados y un breve resumen de los resultados obtenidos.
 5. Conclusiones: resumen de lo realizado y conclusiones técnicas acerca de la práctica y el uso de sistemas de producción (ventajas e inconvenientes).
 6. Comentarios personales: opinión personal.
- Los ficheros se entregarán por Aula Global. Se entregará un único fichero comprimido (.tgz) **utilizando el comando tar** cuyo nombre será el NIA de uno de los autores precedido por la letra *a* . Al descomprimir este fichero se deberá generar un directorio cuyo nombre debe corresponder con el NIA de uno de los autores precedido por la letra *a*. Este directorio debe contener exclusivamente los siguientes ficheros con los **nombres que se especifican**:
- Fichero con el nombre y la dirección de e-mail de los autores, **autores.txt**
 - El fichero con la ontología, **ontologia.clp**
 - El fichero con las reglas para controlar el juego, **servidor.clp**
 - El fichero con las reglas que modelan el comportamiento de los personajes, **cliente.clp**
 - Los ficheros de traza correspondientes a las 5 evaluación del sistema, **traza1.txt, traza2.txt, ... traza5.txt** y los valores utilizados **jugada1.clp, ... jugada5.clp**

Ejemplo: supongamos que el NIA de uno de los autores es 100000000. Por lo tanto, los ficheros a entregar estarán en un directorio con nombre a100000000. Para generar el fichero comprimido que se debe entregar, haremos:

```
tar -czvf a100000000.tgz a100000000/
```

La práctica se debe hacer en grupos de dos personas. En ningún caso se admitirán prácticas de grupos con más de dos alumnos