

# Doble Grado en Ingeniería Informática y Matemáticas

INGENIERÍA DEL CONOCIMIENTO

**Desarrollo de un Sistema Experto**



**UNIVERSIDAD  
DE GRANADA**

Alberto Estepa Fernández - *albertoestep@correo.ugr.es* - 31014191W

*Junio de 2020*

# Índice

<b>1. Resumen del funcionamiento del sistema experto</b>	<b>2</b>
<b>2. Descripción del proceso seguido</b>	<b>3</b>
2.1. Procedimiento seguido para el desarrollo de la base de conocimiento .	3
2.2. Procedimiento de validación y verificación del sistema . . . . .	3
<b>3. Descripción del sistema</b>	<b>4</b>
3.1. Variables de entrada del problema . . . . .	5
3.2. Variables de salida del problema . . . . .	7
3.3. Conocimiento global del sistema . . . . .	7
3.4. Especificación de los módulos desarrollados . . . . .	9
3.5. Hechos y reglas de cada módulo . . . . .	9
<b>4. Breve manual de uso del sistema</b>	<b>14</b>

# 1. Resumen del funcionamiento del sistema experto

El problema que tenemos entre manos consiste en aconsejar al usuario matriculado en el grado de Ingeniería Informática sobre que rama de la carrera escoger o sobre que asignaturas matricularse.

Como vemos necesitamos desarrollar una serie de tareas que requieren razonamiento humano para resolver el problema. Por esto estamos ante el desarrollo de un sistema experto.

El sistema lo hemos implementado en CLIPS. Usaremos la lógica por defecto para que el razonamiento de nuestro sistema.

El funcionamiento del sistema consiste en realizar preguntas al usuario sobre lo que quiere y/o prefiere sobre sus gustos con la informática y sobre sus habilidades en el campo del estudio; y con dicha información el sistema se decantará por uno u otro camino razonando sobre lo que el usuario ha introducido.

Más concretamente, el sistema da la bienvenida al usuario y le pregunta sobre qué funcionalidad quiere asesoramiento: **ramas de especialidad** o **asignaturas de la carrera**. Según la respuesta del usuario el sistema aconsejará sobre uno u otro tema.

Si el tema es **ramas de especialidad** el sistema recogerá información sobre *gusto del usuario por las matemáticas, grado de trabajo del usuario, preferencia del usuario por el hardware o el software, preferencia sobre el futuro laboral del usuario y nota media de la carrera*; pudiendo obviar algunos puntos de esta información ya que el sistema es capaz de razonar y seleccionar las preguntas que le permitan obtener el consejo correcto con la menor información posible.

Por otro lado, si el tema es **asignaturas de la carrera** el sistema recogerá información sobre *preferencia del usuario por el hardware o el software, grado de programación que desea el usuario en las asignaturas que curse y grado de trabajo que requieren las asignaturas que curse*; de nuevo el sistema puede obviar algunas preguntas ya que el sistema es capaz de razonar y seleccionar las preguntas que le permitan obtener el consejo correcto con la menor información posible. Además el sistema es capaz de razonar mediante lógica por defecto para obtener información necesaria para el consejo.

Por último el sistema preguntará al usuario si desea realizar otra consulta, lo que dota al sistema de robustez integrando ambas partes en un mismo sistema.

## 2. Descripción del proceso seguido

En esta sección trataremos el procedimiento seguido para el desarrollo de la base del conocimiento y el procedimiento de validación y verificación del sistema.

### 2.1. Procedimiento seguido para el desarrollo de la base de conocimiento

Para adquirir parte del conocimiento hemos realizado una educación de los requisitos del problema mediante una entrevista con nuestro experto en el conocimiento, donde se han tratado los incidentes críticos (se hizo en la tarea *Asesorar a un alumno de informática en su elección de asignaturas en las que matricularse*). Además hemos extraído conceptos y factores relevantes mediante técnicas rejilla y árboles de decisión (desarrollados en la tarea *Rejilla de Repertorio sobre Asignaturas* y en la tarea *Ejemplo práctico de adquisición del conocimiento (Clase del Viernes 13 de Marzo)*).

Mediante éstas técnicas conseguimos definir qué conceptos son discriminantes para provocar el razonamiento del sistema y poder concluir con los resultados esperados. Más concretamente vimos que es necesario tener información sobre las asignaturas y sus créditos para realizar la elección de asignaturas a matricular, las habilidades del usuario con respecto al estudio y preferencias del mismo hacia las partes de la informática, como puede ser la programación, el nivel de trabajo, preferencias por el software o el hardware, etc.

### 2.2. Procedimiento de validación y verificación del sistema

Todo el procedimiento de validación y verificación del sistema se ha llevado a cabo durante el desarrollo e implementación del sistema. Más concretamente:

- **Validación:** Se ha comprobado en cada módulo que se cumplían las especificaciones de diseño. La comunicación entre el usuario y el sistema es rápida, concreta y eficaz, indicando en cada momento las respuestas validas que tiene que introducir el usuario y comprobando que se introducen correctamente. Además el sistema siempre comunica al usuario el razonamiento relevante que realiza justificando las respuestas.

También se ha pedido a algunos usuarios (alumnos actuales de la carrera) que prueben el sistema para validar y encontrar fallos, resultando efectiva y bien valorada.

Anotamos que si se ha reflejado en la opinión de los usuarios que han realizado la prueba del sistema no incluye la totalidad de asignaturas de la carrera, cosa que se establece con futura mejora del proyecto, pero que para éste trabajo no se considera necesario.

■ **Verificación:**

- Para comprobar que el sistema es *consistente*, es decir, que no se llega a conclusiones incoherentes, por cada regla que se introducía en el sistema, se comprobaba que efectivamente se ejecutaba cuando los hechos y la prioridad existente se correspondía y que no contradecía a ningún otro hecho (en ese caso, se retractaba al momento).
- Para comprobar que el sistema es *preciso*, se ha comprobado minuciosamente en los hechos originales del conocimiento no existiesen errores y se ha sometido un filtro exhaustivo a cada respuesta dada por el usuario en la comunicación usuario-sistema, volviendo a requerir la información válida si ésta no hubiese sido introducida originalmente de forma correcta.
- Y por último para comprobar que el sistema es *completo*, se han estudiado cada una de las posibles combinaciones de respuestas introducidas por el usuario en el módulo de **ramas de especialidad** comprobando que el resultado era correcto y biyectivo con respecto al árbol de decisión desarrollado en la adquisición del conocimiento además de comprobar que todas las ramas son aconsejables mediante una combinación de respuestas y por supuesto, la misma combinación de repuestas involucra el mismo resultado siempre; para el módulo de **asignaturas de la carrera**, se ha comprobado que todas las asignaturas pueden ser recomendadas si se cumplen los requisitos del sistema y de nuevo que la misma combinación de repuestas involucra el mismo resultado. El sistema siempre da un resultado a la consulta pedida, que puede ser más o menos personal al usuario si el usuario introduce más o menos respuestas en el sistema (dado que el usuario puede pedir consejo casi en cualquier momento del sistema).

### 3. Descripción del sistema

En esta sección se incluye la información sobre las variables de entrada y salida del problema y como se representan, así como de los hechos y relaciones que se cargan inicialmente en el sistema como conocimiento. También incluimos la descripción de cada uno de los módulos desarrollados y los hechos y reglas implementados en cada uno.

En los apartados siguientes hablaremos los módulos principales por separado, para favorecer la sencillez de la lectura y de la comprensión, aunque ya sabemos que todos forman una unidad conjunta.

### 3.1. Variables de entrada del problema

Para el módulo **Inicio** la variable de entrada será la respuesta a la pregunta inicial, que indicará el módulo a donde dirigirá el sistema al usuario. Las posibilidades son: **ramas**, para dirigirlo al módulo **ramas**; **asignaturas**, para dirigirlo al módulo **asignaturas** y **salir** para dirigirlo a la salida.

Para el módulo **asignaturas** las variables de entrada serán las asignaturas introducidas por el usuario entre la totalidad de asignaturas del sistema. Se introducirá al conocimiento como:

```
(Asignatura
  (Nombre ?nombre)
  (puntuacion 0)
  (creditos ?numero)))
```

donde **?nombre** y **?numero** dependerá de lo que introduzca el usuario.

También será una variable de entrada, el número de créditos a matricular, que será un valor entero ente 6 y 72 (se han establecido dichos límites para que se ajuste a la realidad). Dicha variable se introducirá al sistema como:

```
(creditos_a_matricular ?respuesta)
```

donde **?respuesta** dependerá de lo que introduzca el usuario.

Y por último, también tendremos como entrada las respuestas del usuario a las preguntas del sistema, que pueden ser:

- Respuesta a la pregunta 1:

```
(prefiere campo_informatica ?respuesta)
```

donde **?respuesta** puede ser hardware o software.

- Respuesta a la pregunta 2a:

```
(prefiere carga_trabajo ?respuesta)
```

donde **?respuesta** puede ser alta o baja.

- Respuesta a la pregunta 2b:

```
(prefiere grado_programacion ?respuesta)
```

donde `?respuesta` puede ser alto o bajo.

Para el módulo **ramas** las variables de entrada serán las respuestas del usuario a las preguntas del sistema, que pueden ser:

- Respuesta a la pregunta 1:

<code>(matematicas ?respuesta)</code>
---------------------------------------

donde `?respuesta` puede ser si o no.

- Respuesta a la pregunta 2a:

<code>(grado_trabajador ?respuesta)</code>
--

donde `?respuesta` puede ser 1, 2 o 3. Aunque incluimos que el usuario pueda introducir la variable siguiente o consejo.

- Respuesta a la pregunta 2b:

<code>(informatica ?respuesta)</code>
---------------------------------------

donde `?respuesta` puede ser software o hardware. Aunque incluimos que el usuario pueda introducir la variable consejo.

- Respuesta a la pregunta 2b-1:

<code>(nota_numerica ?nota)</code>
------------------------------------

donde `?nota` puede ser cualquier valor numérico entre 5 y 10. Aunque incluimos que el usuario pueda introducir la variable consejo y siguiente.

- Respuesta a la pregunta 2b-2:

<code>(trabaja ?respuesta)</code>
-----------------------------------

donde `?nota` puede ser publica o privada. Aunque incluimos que el usuario pueda introducir la variable consejo.

## 3.2. Variables de salida del problema

Las variables de salida que considera nuestro problema son las que producen los dos módulos principales: **asignaturas** y **ramas**

- Variables de salida producidas por **asignaturas**: Los valores de salida proporcionados por el sistema son las asignaturas recomendadas y el motivo de cada recomendación (todo en forma de cadena o *string*). También se proporciona valores de salida (también en forma de cadena o *string*) cuando el sistema utiliza la lógica por defecto para asumir o retractar hechos, indicando el motivo de la realización de cada acción.
- Variables de salida producidas por **ramas**: Los únicos valores de salida proporcionados por el sistema son los posibles consejos (en forma de cadena o *string*) que se producen cuando el usuario introduce el hecho **consejo** o cuando se concluye el razonamiento del sistema llegando a una hoja del árbol de decisión. Cada consejo es personal y depende de las respuestas del usuario.

## 3.3. Conocimiento global del sistema

Al estar el sistema bastante modularizado, realmente no carga inicialmente ningún hecho o regla, si no que se va cargando una vez que se introduce en cada módulo.

Para el módulo **asignaturas**, el conocimiento inicial incluye las 20 asignaturas de segundo y de tercero del grado de Ingeniería Informática (de tercero del segundo cuatrimestre sólo se han introducido las asignaturas de la rama de CSI), además de la asignatura prácticas de empresa (ésta se ha introducido para comprobar que el sistema trata correctamente asignaturas de diferentes créditos). Cada una de las 21 asignaturas se ha introducido como:

```
(asignatura pdoo)
(asignatura scd)
(asignatura so)
(asignatura ed)
(asignatura ec)
(asignatura fis)
(asignatura alg)
(asignatura fdb)
(asignatura ia)
(asignatura ac)
(asignatura ig)
(asignatura ddsi)
(asignatura mc)
(asignatura fr)
(asignatura ise)
(asignatura aa)
(asignatura ic)
(asignatura mca)
(asignatura mh)
(asignatura tsi)
(asignatura practicas)
```



Para introducir más conocimiento sobre otras asignaturas, solo habría que incluir el nombre ahí.

Por otro lado, el sistema razonará por defecto indicando que todas las asignaturas tienen 6 créditos (debido a que es lo habitual). Salvo en el caso de la asignatura **practicas**, que sabemos que son 12 créditos. Esto se ha introducido en el conocimiento:

```
(creditos practicas 12)
```

Si quisiéramos cambiar el número de créditos de cualquier asignatura simplemente habría que asertar el hecho (**creditos** 'nombre' 'número'), y el sistema retractaría los hechos por defecto automáticamente.

Por último el sistema razonará por defecto indicando que todas las asignaturas poseen mas parte de Software que de Hardware, que todas las asignaturas poseen alto grado de programación y que todas las asignaturas poseen alto nivel de trabajo debido a que esto suele ser habitual. Como es lógico existen excepciones que hemos introducido como:

```
;; mas parte de Hardware que de Software
(campo_informatica scd hardware seguro)
(campo_informatica ec hardware seguro)
(campo_informatica ac hardware seguro)
(campo_informatica ise hardware seguro)
(campo_informatica fr hardware seguro)
(campo_informatica ed hardware seguro)

;; bajo nivel de trabajo
(carga_trabajo fis baja seguro)
(carga_trabajo ddsi baja seguro)
(carga_trabajo scd baja seguro)
(carga_trabajo fdb baja seguro)
(carga_trabajo alg baja seguro)
(carga_trabajo ac baja seguro)

;; bajo grado de óprogramacin
(grado_programacion so bajo seguro)
(grado_programacion fdb bajo seguro)
(grado_programacion ddsi bajo seguro)
(grado_programacion mca bajo seguro)
(grado_programacion mc bajo seguro)
```

De igual manera si quisiéramos modificar el conocimiento habría que introducir o eliminar alguna de las reglas en dicho formato.

Para el módulo **ramas**, el conocimiento inicial son las cinco ramas de especialidad de la carrera de Ingeniería Informática de la UGR. No se han introducido en hechos como tal debido a la simplicidad del sistema.

### 3.4. Especificación de los módulos desarrollados

El sistema implementado está estructurado en dos grandes módulos: **ramas** y **asignaturas**. Cada una de ellas se corresponden con una funcionalidad extraída de los requisitos funcionales definidos en la fase de identificación del problema.

El módulo **asignaturas**, cuyo objetivo es asesorar al usuario sobre que asignaturas escoger entre las dadas por el mismo usuario dentro de un conjunto de posibles asignaturas, utiliza el conocimiento explicado en la sección 3.3 y el conocimiento que se deduce es aquel para lo que ha sido diseñado: recomendar al usuario las asignaturas, produciendo las variables explicadas en la sección 3.2.

De forma análoga, el módulo **ramas**, cuyo objetivo es asesorar al usuario sobre que rama de especialidad elegir, también utiliza el conocimiento explicado en la sección 3.3 y de nuevo el conocimiento que se deduce es aquel para lo que ha sido diseñado: recomendar al usuario la rama de especialidad que debería cursar, produciendo las variables explicadas en la sección 3.2.

Para que el usuario pueda introducirse en cualquiera de los módulos se ha introducido otro módulo **Inicio** cuya única función es derivar en alguno de los módulos anteriores una vez el usuario haya contestado la pregunta inicial. No utiliza ningún conocimiento salvo el introducido por el usuario en la respuesta a la pregunta que formula y deriva en alguno de los módulos anteriores para deducir el conocimiento pedido.

También disponemos de un módulo de salida **salir**, para salir del sistema y un módulo donde se realizan las comprobaciones de las preguntas (**preguntas**) y que es exportado a los módulos **ramas** y **asignaturas** (realmente la motivación de la creación de dichos módulos es meramente la sencillez del código, por lo que no creemos conveniente darle mayor explicación. Si se desea, el código incluido en el fichero **P2.clp** está correctamente comentado e incluye más información al respecto).

### 3.5. Hechos y reglas de cada módulo

Incluimos todas las reglas que utiliza el sistema durante su funcionamiento, divididas por módulos:

- **Inicio:**
  - **Bienvenido\_inicio:** Menú inicial para derivar al usuario a la sección de **ramas** o **asignaturas** o salir del sistema
- **ramas:**

- Bienvenido: Menú informativo.
  - Pregunta\_1: Pregunta sobre gusto de matemáticas.
  - Pregunta\_2: Pregunta sobre grado de trabajo.
  - no\_contesta\_pregunta\_2: Produce un salto en el árbol si el usuario no quiere contestar a la anterior pregunta.
  - Convertir\_grado\_trabajador\_mucho, Convertir\_grado\_trabajador\_medio y Convertir\_grado\_trabajador\_poco: Convierte las variables introducidas por el usuario a hechos usables por el sistema.
  - Pregunta\_3: Pregunta sobre preferencia en Hardware o Software.
  - Pregunta\_4: Pregunta sobre calificación media.
  - comprobar\_respuesta\_nota: Comprueba que la respuesta a la anterior pregunta es correcta.
  - comprobar\_respuesta\_nota\_consejo: Comprueba que la respuesta a la anterior pregunta no es pedir consejo.
  - comprobar\_respuesta\_nota\_siguiente: Comprueba que la respuesta a la anterior pregunta no es saltar de pregunta.
  - no\_contesta\_pregunta\_4: Produce un salto en el árbol si el usuario no quiere contestar a la anterior pregunta.
  - comprobar\_respuesta\_nota\_fallo: Si no ha introducido ningún valor correcto, vuelve a preguntar.
  - comprobar\_respuesta\_nota\_fallo\_rango: Si ha introducido valor correcto pero no en el rango adecuado, vuelve a preguntar.
  - Convertir\_nota\_alta, Convertir\_nota\_media y Convertir\_nota\_baja: Convierte las variables introducidas por el usuario a hechos usables por el sistema.
  - Pregunta\_5: Pregunta sobre preferencia en empresa pública o privada.
  - hoja1, hoja2, hoja3, hoja4, hoja5, hoja6, rama1, rama2, rama2\_1, rama2\_2 y consejo\_inicial: Introducen diferentes consejos según la posición del árbol de decisión en el que se encuentren.
  - cambiar\_modulo: Permite volver a modulo Inicio
- asignaturas:
- Elige\_asignaturas: Pregunta de elección de asignaturas.
  - assert\_asignaturas: Introduce las asignaturas elegidas en el conocimiento.
  - borrar\_eleccion: Borra el hecho de control de elegir asignaturas.
  - Comprobar\_asignaturas\_incorrecta: Borra la elección de la asignatura si no está en el conocimiento.

- `retracta_creditos`: Retracta los créditos que se contradigan entre las asignaturas del conocimiento.
- `Comprobar_asignaturas_correcta`: Comprobación de que la asignatura es correcta.
- `Pregunta_creditos`: Pregunta sobre los créditos a matricular.
- `Comprobar_creditos_incorrecta`: Comprueba que los créditos elegidos estén entre el rango especificado.
- `Comprobar_creditos_correcta`: Comprueba que los créditos elegidos estén entre el rango especificado.
- `comprobar_creditos_insuficientes`: Comprueba haya más asignaturas que créditos. Caso negativo.
- `creditos_insuficientes`: Al darse el caso negativo, vuelve a solicitar créditos.
- `eliminar_asignaturas`: Al darse el caso negativo, elimina las asignaturas guardadas en el conocimiento.
- `eliminar_eliminar_asignaturas`: Elimina el hecho de control de eliminar asignaturas.
- `comprobar_creditos_suficientes`: Comprueba haya más asignaturas que créditos. Caso positivo.
- `creditos_suficientes`: Permite avanzar a la fase de preguntas.
- `eliminar_comprobados`: Elimina el hecho de control de las asignaturas que no se han comprobado.
- `eliminar_eliminar_comprobados`: Elimina el hecho de control de eliminar el hecho de control de las asignaturas que no se han comprobado.
- `software_por_defecto`: Aplica la lógica por defecto a las asignaturas de software.
- `retracta_software_por_defecto`: Retracta la lógica por defecto a las asignaturas de software.
- `Pregunta_1`: Pregunta sobre preferencia en Hardware o Software.
- `consecuencias_pregunta_1`: Aporta valor a aquellas asignaturas que se correspondan con la pregunta anterior.
- `carga_de_trabajo_alta_por_defecto`: Aplica la lógica por defecto a las asignaturas de carga de trabajo alta.
- `retracta_carga_de_trabajo_alta_por_defecto`: Retracta la lógica por defecto a las asignaturas de carga de trabajo alta.
- `Pregunta_2`: Pregunta sobre preferencia en carga de trabajo.
- `consecuencias_pregunta_2`: Aporta valor a aquellas asignaturas que se correspondan con la pregunta anterior.

- `consecuencias_pregunta_2_sin_casos`: Como es rama del árbol de decisión, incluye el hecho de control consejo si no hay asignaturas que se correspondan con la pregunta anterior.
- `grado_programacion_alto_por_defecto`: Aplica la lógica por defecto a las asignaturas de nivel de programación alto.
- `retracta_grado_programacion_alto_por_defecto`: Retracta la lógica por defecto a las asignaturas de nivel de programación alto.
- `Pregunta_3`: Pregunta sobre preferencia en nivel de programación.
- `consecuencias_pregunta_3`: Aporta valor a aquellas asignaturas que se correspondan con la pregunta anterior.
- `consecuencias_pregunta_3_sin_casos`: Como es rama del árbol de decisión, incluye el hecho de control consejo si no hay asignaturas que se correspondan con la pregunta anterior.
- `max_puntuacion`: Calcula el máximo entre varias puntuaciones de asignaturas.
- `eliminar_fase_preguntas`: Elimina el hecho de control fase preguntas.
- `consejo`: Incluye consejo mientras no se completen los créditos.
- `consejo_ultima`: Incluye último consejo porque se ha llegado al límite de créditos.
- `consejo_pasado`: Incluye último consejo porque se ha superado el límite de créditos.
- `recomendar`: Recomendamos asignaturas.
- `eliminar_recomendar`: Eliminar el hecho de control de recomendar asignaturas.
- `cambiar_modulo`: Permite volver a modulo Inicio

Incluimos todos los hechos que utiliza el sistema durante su funcionamiento, divididos por módulos:

■ **ramas:**

- `Bienvenido`: Hecho de control inicial.
- `consejo`: Hecho de control para pedir consejo.
- `matematicas ?respuesta`: Respuesta a la pregunta de matemáticas. Puede ser si o no.
- `no_contesta_pregunta_2`: Hecho de control para saltar en el árbol la pregunta 2.
- `grado_trabajador ?respuesta`: Respuesta a la pregunta de grado de trabajo. Puede ser 1, 2 o 3.
- `trabajador mucho`: Convertido el valor 3 de la respuesta anterior a un valor usable por el sistema.

- `trabajador_medio`: Convertido el valor 2 de la respuesta anterior a un valor usable por el sistema.
  - `trabajador_poco`: Convertido el valor 1 de la respuesta anterior a un valor usable por el sistema.
  - `informatica ?respuesta`: Respuesta a la pregunta sobre preferencia entre Hardware y Software. Puede ser software o hardware.
  - `comprobar_respuesta ?nota`: Hecho de control para comprobar la respuesta de la pregunta 4.
  - `nota_numerica ?nota`: Respuesta comprobada a la pregunta sobre nota media. Puede ser cualquier valor numérico entre 5 y 10.
  - `no_contesta_pregunta_4`: Hecho de control para saltar en el árbol la pregunta 4.
  - `nota ?valor`: Respuesta discretizada a la pregunta sobre nota media. Puede ser alta, media o baja.
  - `trabaja ?respuesta`: Respuesta a la pregunta sobre preferencia laboral. Puede ser pública o privada.
  - `cambiar_modulo`: Hecho de control para cambiar módulo.
- `asignaturas`:
- No incluimos los comentados en la sección 3.3.
  - `elegidas`: Hecho de control para comprobar si están las asignaturas elegidas.
  - `eleccion ?respuesta`: Lista de asignaturas elegidas.
  - `ComprobarEleccion ?nombre`: Hecho de control para comprobar si la asignatura es correcta.
  - `Asignatura (Nombre ?nombre) (puntuacion ?p) (creditos ?n)`: Asignatura introducida por el usuario.
  - `creditos_no_contados ?n`: Hecho de control que indica que la asignatura de nombre ?n no ha sido contada todavía para los créditos.
  - `comprobar_creditos ?respuesta`: Hecho de control para comprobar que los créditos elegidos estén entre 6 y 72.
  - `comprobar_creditos_suficientes ?respuesta`: Hecho de control para comprobar las asignaturas suman los créditos pedidos.
  - `no_suficientes_creditos`: Hecho de control que indica que no tenemos suficientes créditos.
  - `eliminar asignaturas`: Hecho de control que indica que debemos eliminar las asignaturas.
  - `eliminar_comprobados`: Hecho de control que indica que debemos eliminar los hechos de control `creditos_no_contados ?n`.

- **fase preguntas:** Hecho de control que indica que entramos en la fase de preguntas.
- **prefiere campo\_informatica ?respuesta:** Indica la preferencia del usuario. Puede ser software o hardware.
- **explicacion ?nombre ?expl:** Incluye la explicación **?expl** sobre la asignatura **?nombre**.
- **prefiere carga\_trabajo ?respuesta:** Indica la preferencia del usuario. Puede ser alta o baja.
- **prefiere grado\_programacion ?respuesta:** Indica la preferencia del usuario. Puede ser alto o bajo.
- **puntuacion\_max ?max:** Indica la información de la asignatura con mejor valor a recomendar.
- **recomendar ?nombre:** Indica la asignatura con mejor valor a recomendar.
- **cambiar\_modulo:** Hecho de control para cambiar módulo.

## 4. Breve manual de uso del sistema

Si quiere ejecutar el sistema implementado en CLIPS deberá cargar el fichero en el directorio con `(load P2.clp)`, y aplicar `(reset)` y `(run)`. A continuación el sistema le dará la bienvenida con un mensaje y un menú y usted como usuario deberá responder con alguna de las tres posibles respuestas: **asignaturas**, **ramas** o **salir**.

- Si elige **salir**, el programa terminará.
- Si elige **asignaturas**, el sistema le conducirá por el modulo **asignaturas**. Aclaremos que en cada uno de los pasos el sistema realizará una comprobación de la respuesta del usuario y si es incorrecta se le pedirá de nuevo que introduzca un nuevo valor.:
  1. El siguiente paso a realizar por el usuario será elegir entre un conjunto de asignaturas el subconjunto entre las que el usuario tiene dudas sobre cuales matricularse.
  2. A continuación el usuario tendrá que introducir un número entre 6 y 72 para indicar los créditos que va a matricularse.
    - a) Una vez respondido, si la suma de los créditos de las asignaturas seleccionadas pueden sumar un valor igual o superior al número de créditos introducido, empezaremos en la fase de preguntas. Si no ocurre esto, se volverán a pedir de nuevo asignaturas entre las que elegir.
  3. La primera pregunta será sobre su preferencia para el software o el hardware y el usuario deberá responder **software** o **hardware**.

- a) Si la respuesta introducida es **hardware**, la siguiente pregunta será sobre la preferencia en la carga de trabajo y el usuario deberá responder **alta** o **baja**, o **consejo** si el usuario no desea responder a la pregunta y pedir ya el consejo.
    - b) Si la respuesta introducida es **software**, la siguiente pregunta será sobre la preferencia de un grado de programación en las asignaturas y el usuario deberá responder **alto** o **bajo**, o **consejo** si el usuario no desea responder a la pregunta y pedir ya el consejo.
  - 4. Como conclusión, el sistema le expondrá las asignaturas, entre el subconjunto elegido, que deberá de matricularse según sus respuestas y los créditos a matricular; argumentando su elección.
  - 5. El sistema le dirigirá al menú inicial de nuevo.
- Si elige **ramas**, recibirá un mensaje de bienvenida al modulo **ramas** y una serie de explicaciones. Aclaremos que en cada uno de los pasos el sistema realizará una comprobación de la respuesta del usuario y si es incorrecta se le pedirá de nuevo que introduzca un nuevo valor.:
    - 1. La primera pregunta será sobre si le gustan las matemáticas y el usuario tendrá que responder **si** o **no**, o **consejo** si el usuario no desea responder a la pregunta y pedir ya el consejo.
      - a) En el caso que responda **si**, el sistema le preguntará sobre si el usuario es trabajador y dicho usuario tendrá que responder con 1, 2 o 3. Podrá responder también con **consejo** si el usuario no desea responder a la pregunta y pedir ya el consejo o con **siguiente** si no desea responder a la pregunta.
      - b) En el caso que responda **no**, el sistema le preguntará sobre si el usuario prefiere el software o el hardware y dicho usuario tendrá que responder con **software** o **hardware**. Podrá responder también con **consejo** si el usuario no desea responder a la pregunta y pedir ya el consejo.
        - 1) Si la respuesta es **software**, el sistema le preguntará sobre la calificación media del usuario en la carrera pudiendo ser respondida con cualquier número entre 5 y 10. Podrá responder también con **consejo** si el usuario no desea responder a la pregunta y pedir ya el consejo o con **siguiente** si no desea responder a la pregunta.
        - 2) En el caso que responda **hardware**, el sistema le preguntará sobre si el usuario prefiere trabajar en la empresa pública o en la empresa privada y dicho usuario tendrá que responder con **pública** o **privada**. Podrá responder también con **consejo** si el usuario no desea responder a la pregunta y pedir ya el consejo.
    - 2. Como conclusión, el sistema le indicará la rama de especialidad aconsejada con su respectiva explicación.
    - 3. El sistema le dirigirá al menú inicial de nuevo.