

# Desarrollo de un Sistema Basado en el Conocimiento

## Funcionamiento del sistema:

El sistema incluye un tipo de razonamiento con incertidumbre usado anteriormente en esta misma práctica, que es el de factores de certeza.

La primera parte se ha podido ver previamente en la práctica anterior y en el primer apartado de esta segunda, pero su funcionamiento es lo suficientemente simple como para volver a echarle un vistazo. A partir de la combinación de hechos y reglas, el sistema interactúa con el usuario de la forma más clara y escueta posible que le permita llegar a una conclusión coherente sobre la elección de rama en la carrera. Su funcionamiento parte de la pregunta directa sobre la asignatura favorita tanto de primero como de segundo para saber cuáles son los gustos del alumno, posteriormente se informa acerca de la nota media de este y por último obtiene la información, aunque un poco subjetiva, relacionada con el esfuerzo del estudiante en la carrera.

A raíz de esta información el sistema ya es capaz de decidir por qué parte empezar a encaminar al usuario y, conforme este vaya respondiendo preguntas relacionadas con el camino que le muestra el sistema, nuestro SBC irá desplegando nuevas preguntas o decisiones internas que le conduzcan a una conclusión final. Cuando esta es alcanzada, simplemente se muestra al usuario, acompañada de una justificación (algo menos exhaustiva que la del apartado B) y de un experto que realiza la recomendación.

Para la realización del apartado B como un complemento del A, las modificaciones que ha sufrido nuestro sistema es el punto de partida y la adición de algunos módulos extras.

En cuanto al punto de partida, el sistema ofrece la opción de empezar por la parte que te aconseja sobre la rama, sobre la que te ayuda a decidir entre dos asignaturas o por ninguna directamente y dar por terminado el uso del sistema. Considerando que para el apartado B el usuario selecciona la opción de la ayuda sobre dos asignaturas, el sistema muestra por pantalla 10 asignaturas aleatorias seleccionadas de entre las de primero y segundo que, aunque no son todas las que podría necesitar un sistema completo, sirven como una muestra con la que comprobar de una forma más que suficiente el correcto funcionamiento de este.

Una vez que estas asignaturas son mostradas por pantalla, el usuario selecciona dos de ellas entre las que tenga dudas y el sistema comienza con una pequeña recolección de información para sacar conclusiones. Como para esta parte se utilizan factores de certeza, lo que hace el sistema es realizar 4 preguntas sobre cada una de las dos asignaturas que seleccione el usuario para saber su postura con respecto a ellas desde puntos de vista que el estudiante no podría haber considerado. Estas respuestas son consideradas como Evidencias de 'sí' o 'no' con una certeza de 1 que se transforman automáticamente en Factores de Certeza. Estos factores podrán disparar unas reglas lógicas definidas en el sistema con una certeza asociada y para terminar se considera la certeza de cada asignatura, siendo seleccionada aquella que mayor valor tenga.

Como punto extra, al terminar cualquiera de los dos apartados, el sistema planteará la opción de dar por terminada la consulta, reiniciar la recién realizada o pasar al apartado opuesto al seleccionado, así en la misma ejecución podremos probar tantas combinaciones como queramos.

## Funcionamiento detallado por pasos del apartado A

- Se muestran las 10 asignaturas del primer curso por pantalla (MP, FP, IES, TOC, CAL, CALEM, ES, FS, FFT y LMD)
- Se valida que la respuesta sea correcta y en caso de no serlo, se repite la introducción de los datos hasta que lo sea
- Se muestran las 10 asignaturas del segundo curso por pantalla (ED, AC, EC, SCD, FBD, ALG, SO, PDOO, FIS y IA)
- Se valida que la respuesta sea correcta y en caso de no serlo, se repite la introducción de los datos hasta que lo sea
- Se pregunta por la nota media del expediente del alumno (SUFICIENTE (5), BIEN (6), NOTABLE (7-8), SOBRESALIENTE (9-10))
- Se valida que la respuesta sea correcta y en caso de no serlo, se repite la introducción de los datos hasta que lo sea
- Se pregunta por el esfuerzo que el alumno considera que realiza en la carrera (ALTO/MEDIO/BAJO)
- Se valida que la respuesta sea correcta y en caso de no serlo, se repite la introducción de los datos hasta que lo sea
- A partir de las asignaturas favoritas decide la rama por la que empezar preguntando al usuario:
  - o En caso de haber seleccionado dos asignaturas de la misma rama -> Empieza por ella sin duda
  - o En caso de haber seleccionado una asignatura de cada rama -> se basa en la media para elegir entre las dos menciones, priorizando medias altas (De haber seleccionado una de CSI y otra de TI, pero con media alta, empieza por CSI. Sin embargo, si elige una de IC y otra de IS, pero la media es baja, empieza por IC)
- Se realizan las preguntas relacionadas con la rama seleccionada de comienzo y se almacena la rama por la que empezó preguntando.
  - o De haber respuesta positiva en todas las preguntas -> Se aconseja esa rama
  - o De haber una mayoría de respuestas positivas -> Pregunta extra que confirme si va o no
  - o De no haber respuestas positivas
    - Se dispara una pregunta que te reconduzca por otro camino (Por ejemplo, si te gusta más el hardware que el software pasa a preguntar por IC, pero si es al revés, preguntará por alguna otra)
    - Te reconduce a otra rama directamente según considere por la media y el esfuerzo
  - o De no haber respuesta positiva en ninguna rama que ayude a aconsejar con sentido, recomienda la rama por la que empezó (se almacenó) porque es una decisión con criterio.
- EL sistema pregunta si quieres darlo por terminado, volver a empezar o pasar al apartado B y hacer consultas sobre asignaturas.

## Funcionamiento detallado por pasos del apartado B

- Se muestran las 10 asignaturas por pantalla (MP, IA, IES, TOC, CAL, ES, AC, EC, FBD y PDOO)
- Se valida que la respuesta sea correcta y en caso de no serlo, se repite la introducción de los datos hasta que lo sea
- Se muestran las 9 asignaturas restantes por pantalla
- Se valida que la respuesta sea correcta y diferente a la seleccionada en el paso 1 para evitar comparar asignaturas con ellas mismas. Si los datos son inválidos, ya sea porque no es ningún elemento de la lista o porque sea una asignatura ya seleccionada, se fuerza a una reintroducción de estos hasta que sean aceptados.
- Se disparan 8 preguntas, 4 por cada asignatura que serán respondidas con 'sí' o 'no'
- Se valida que en el paso anterior se haya introducido un 'sí' o un 'no' haciendo que el usuario repita su respuesta en caso contrario.
- Se disparan las reglas necesarias internamente a raíz de las respuestas del estudiante generando unas hipótesis con ciertas certezas
- Se procesan los datos combinando y encadenando los necesarios
- Se muestran todas las hipótesis con su certeza por pantalla
- Se muestra la mejor hipótesis
- Se muestra el razonamiento seguido para concluir que es la mejor hipótesis.
- EL sistema pregunta si quieres darlo por terminado, volver a empezar o pasar al apartado A y hacer consultas sobre ramas.

## Descripción del proceso seguido apartado B:

- En cuanto al **desarrollo** de la base de conocimiento hemos hecho uso del ejemplo proporcionado por el profesor en la plataforma de la asignatura relacionado con los factores de certeza.  
([https://pradogrado2021.ugr.es/pluginfile.php/556076/mod\\_resource/content/3/Ejemplo%20de%20uso%20de%20factores%20de%20Certeza%20en%20CLISP.pdf](https://pradogrado2021.ugr.es/pluginfile.php/556076/mod_resource/content/3/Ejemplo%20de%20uso%20de%20factores%20de%20Certeza%20en%20CLISP.pdf)).

Tras esto, se ha hecho una selección de solamente algunas asignaturas que ofertar al usuario para que seleccione entre cuales salir de dudas, ya que para poder ayudarlo necesitamos realizar preguntas relacionadas a cada una y tocarlas todas es casi imposible. Cuando seleccionamos estas asignaturas (MP, IA, IES, TOC, CAL, ES, AC, EC, FBD y PDOO), definimos la información que nos iba a hacer falta sobre cada una de ellas que, en nuestro caso, son 4 preguntas sobre el contenido y herramientas que se vean en cada una.

El siguiente paso es definir las reglas necesarias, ya sean independientes o combinadas, para que el sistema termine asignando una certeza diferente a cada asignatura y con esto decidir cuál es la mejor opción. Para esta parte en la que las reglas introducen certezas en las hipótesis se hace uso de las funciones 'encadenado' y 'combinación' proporcionadas en PRADO.

Se ha reutilizado la parte del código propio del ejercicio sobre Factores de Certeza del apartado 2 de esta práctica que mostraba por pantalla todas las hipótesis junto a su certeza, seleccionaba la mayor de ellas y enseñaba la justificación del razonamiento.

- Para la validación y verificación seguiremos nuestro propio ejercicio de teoría sobre el tema 6 de teoría.  
([https://pradogrado2021.ugr.es/pluginfile.php/556092/mod\\_resource/content/2/Verificaci%C3%B3n%20y%20Validaci%C3%B3n%20de%20SBC.pdf](https://pradogrado2021.ugr.es/pluginfile.php/556092/mod_resource/content/2/Verificaci%C3%B3n%20y%20Validaci%C3%B3n%20de%20SBC.pdf))

Lo primero será la verificación que comprueba que el sistema cumpla con sus requisitos funcionales y no funcionales. Esto lo sabemos por nuestra propia implementación, pues el objetivo es que el sistema te aconseje sobre una rama concreta o una asignatura entre dos que generen dudas y el sistema lo cumple estrictamente. Igual no es totalmente completo, pero para el nivel al que se realiza podemos contar con que satisface unos mínimos. Además de este criterio, esperamos obtener consistencia y corrección antes de pasar a la validación.

El siguiente paso sería la validación que contaría con diferentes pasos y etapas:

1º Probar casos genéricos de ejecución, como son aquellos con respuestas aleatorias combinando 'sí' y 'no'. Se había planteado la opción inicial de realizar todas las comprobaciones posibles para cerciorarnos de que las reglas se disparan como y cuando deben, pero al ser 10 asignaturas, podemos hacer 45 combinaciones diferentes, en las que cada una tendría 8 preguntas con dos valores posibles, lo que hace un total de 11520 casos. Es por esto por lo que el número de comprobaciones se limitará a 20 ejecuciones diferentes y aleatorias.

2º De haber ido todo como se esperaba en el primer paso, haremos ejecuciones que impliquen casos delicados y extremos, como podría ser la introducción de 'no' 8 veces o 'sí' de la misma manera. El riesgo con casos como estos es que no se dispare ninguna regla o que el disparado de todas saque resultados incoherentes o inconsistentes. Otros casos extremos podrían ser negar todas las preguntas de una asignatura y afirmar todas las de la otra o viceversa.

3º Teniendo en cuenta que el sistema debe permitir la modificación y adición de nuevo conocimiento, no podemos conformarnos con que las ejecuciones de los dos primeros pasos hayan sido exitosas, pues podemos haber pasado por partes del sistema que funcionen correctamente y haber ignorado casualmente partes que contengan errores. Es por esto por lo que se comprobará el funcionamiento de cada módulo por separado. Para esto, tanto en la construcción del sistema como una vez finalizado comentando parte de este, comprobamos que independientemente cada módulo haga lo que se le requiere. Primer se validarían aquellos independientes que realicen alguna acción sin la necesidad de otro previo y cuando estos funcionen, comprobaremos progresivamente aquellos que requieran de información de otros módulos para realizar alguna acción.

4º La desventaja que teníamos con respecto a las comprobaciones, que era la elevada cantidad de estas que podíamos hacer, ahora juega a nuestro favor. Esto se debe a que para el análisis de la sensibilidad del sistema seleccionaremos algunos ejemplos concretos y modificaremos únicamente una de las ocho respuestas que se le proporcionan al sistema, debiendo obtener así resultados similares a los que se obtenían sin modificar nada u obteniendo resultados alejados, pero con una justificación que lo respalde. Esta comprobación también se realizará con 20 ejecuciones diferentes.

5º Las pruebas de campo en paralelo con un experto en este caso son innecesarias porque el experto somos nosotros mismos, pero una actividad interesante puede ser intentar predecir el resultado de una combinación antes de ejecutarla para comprobar que el sistema realiza un razonamiento al menos similar al nuestro. Por otra parte, una alternativa a esta sería que un compañero con la misma experiencia en este campo realice algunas pruebas actuando como experto con el fin de encontrar puntos de conflicto que nos ayude a encontrar incongruencias.

## Descripción del sistema

En cuanto al apartado A, es un Sistema Experto para recomendar qué rama elegir a un alumno de Ingeniería Informática. Este está implementado en CLIPS y simula el proceso por el que asesoraríamos a un alumno que nos pregunta a nosotros sobre la rama que elegir, respetando las preguntas y el orden que seguiríamos en el proceso. Este sistema razona y toma decisiones tal y como lo haríamos nosotros y muestra los motivos por los que aconseja esa o esas ramas conservando una relación cercana con aquellos motivos que nos conducirían a nosotros a esa decisión.

Algunos aspectos en los que se ha sido especialmente cauteloso son la sencilla y cómoda gestión de la entrada y salida de datos, una salida clara y precisa, la mayor naturalidad posible en la interacción, la diferenciación de cada parte del sistema para facilitar su entendimiento y editabilidad, capacidad de respuesta ante entradas de 'no sé' y una explicación acorde a cada caso de los que pueda darse.

- Variables de entrada

La entrada de datos se realiza por la interfaz de CLIPSIDE con la forma de respuesta a las preguntas que realiza el sistema. A nivel código, esta interacción es posible por el comando (read). Las variables de entrada para este apartado son respuestas personales sobre el alumno a preguntas como las asignaturas favoritas, el gusto por algún contenido o herramienta, la nota media o el esfuerzo para facilitar que el sistema llegue a una conclusión.

La representación de estas entradas en el sistema consiste en la creación de un hecho que almacene la respuesta. Este hecho debe ser representativo y descriptivo, intentando minimizar la cantidad de hechos necesarios para el funcionamiento del sistema.

- Variables de salida

En cuanto a la salida del sistema, esta es el consejo del experto acompañado de una explicación sobre la decisión y el nombre del experto que la realizó. De nuevo, su representación es un hecho creado a partir de la información introducida, que contiene toda la información necesaria y que será expuesta por la interfaz con la orden (printout). Además, durante el proceso de interacción con el usuario, también muestra preguntas como salidas del sistema, pero no al nivel de la respuesta final.

- Conocimiento del sistema

Inicialmente el sistema parte con un defefacts llamado 'Ramas' que contiene hechos con el nombre de cada rama posible. Estos hechos tienen la estructura de (Ramas [NOMBRE]) y serán usados para mostrar el nombre de la rama aconsejado como resultado de la decisión.

Por otra parte, cuenta con otra estructura de facts llamada 'asignaturas' que almacena las asignaturas tanto de primero como de segundo para poder mostrar el listado de estas en la pregunta de elección para el alumno por la que empieza ese sistema. La representación de estos hechos es (primero|segundo [NOMBRE\_ASIGNATURA]). Por último, incluido en el de facts de 'asignaturas' se ha creado un hecho del tipo (Consejo NADA) útil para el disparado de algunas reglas al final del proceso según exista o no un consejo. Más detalladamente, si al terminar las preguntas no se ha creado un hecho (Consejo ?rama ?explicacion ?experto) que reemplace a esta original, entenderemos que el sistema no ha considerado que el alumno encaje con ninguna y vuelve a la primera que decidió como punto de partida porque es de la única que tiene algo de base para la elección.

- Módulos desarrollados

La modularización llevada a cabo es aquella que hace uso de estructuras 'defmodule'. Contábamos con la alternativa de modularización con hechos de control o la asignación de la misma prioridad a todas las reglas de una fase, siendo esta mayor que las que correspondan con una fase posterior, pero por temas de facilidad y documentación la elección ha sido utilizar la herramienta que proporciona el propio CLIPS para definir módulos.

○ *Estructura en módulos.* Módulos definidos en el sistema:

- DeclaracionRamas
- PreguntasPrevias
- DecisionPrimerCamino
- ModuloControlDeBloque
- ModuloIC
- ModuloIS
- ModuloCSI
- ModuloSI
- ModuloTI
- VolvemosAPrimera
- Decision

○ *Descripción de los módulos.*

- DeclaracionRamas: Módulo que contiene únicamente el de facts 'Ramas'. Su objetivo es hacer accesible desde cualquier módulo los hechos de partida (conocimiento inicial del sistema) a través de un export. No hace uso del conocimiento con objetivos deductivos ni se deduce conocimiento nuevo.
- PreguntasPrevias: Módulo que realiza las preguntas iniciales al usuario para la elección posterior de la primera rama por la que conducir al estudiante. Su objetivo es recolectar la información que considere necesaria para tomar una decisión prematura sobre la rama a la que el usuario parece estar orientado. Esta información será transmitida al módulo de 'DecisionPrimerCamino' a través de un export. No hace uso del conocimiento inicial del sistema, pero sí que deduce el conocimiento sobre sus asignaturas preferidas ((favorita\_primero ?fav) y (favorita\_segundo ?fav), útiles para la decisión de partida) y de su media y esfuerzo ((nota\_media ?nota) y (esfuerzo ?esfuerzo), útiles a lo largo de todo el programa)

- **DecisionPrimerCamino:** Módulo que utiliza las respuestas a las 4 primeras preguntas que se le hacen al estudiante para decidir cual es la rama de inicio siguiendo el criterio anterior:
  - Si ambas asignaturas eran de la misma rama, empezará por ella
  - Si había dos ramas diferentes entre sus respuestas, se decantará por la que menos media necesite si su media es baja y por la que más media necesite si es alta.

Esta decisión será almacenada dos veces, una que se eliminará cuando dejemos de preguntar por esa rama y otra que se mantendrá estática en el sistema para el caso en el que no encuentre una rama adecuada, basarse en la primera que decidió que podía ser factible. El conocimiento que utiliza es el creado en el módulo PreguntasPrevias a partir de las respuestas del usuario y el conocimiento que se deduce es tanto el de la rama de comienzo como el resultado al que recurriremos en caso de ser necesario.

- **ModuloControlDeBloque:** Módulo que utiliza el conocimiento deducido en el de DecisionPrimerCamino para decidir a qué módulo dirigir el flujo del programa, comprobando no haber pasado por esta ya con el fin de esquivar bucles o preguntas repetidas. El conocimiento que utiliza es aquel importado de los módulos 'DecisionPrimerCamino' y 'DeclaracionRamas', por lo que es el primero que utiliza conocimiento del que existe inicialmente en el sistema relacionado con las ramas. Esto es debido a que conforme comprobemos si el usuario va hacia una rama, esta será eliminada de los hechos para que no pasamos varias veces por la misma línea de razonamiento sin salida. No se deduce conocimiento porque únicamente se utiliza este para elegir el módulo a meter en la pila y no para crear hechos de ningún tipo.
- **ModuloIC:** El objetivo de este módulo y de los cuatro siguientes son prácticamente el mismo, que es realizar las preguntas relacionadas con esta rama y tomar la decisión de aconsejársela definitivamente o derivar a otra si no concuerda del todo. Las preguntas son reglas que se disparan secuencialmente provocando la activación de alguna de las siguientes según la respuesta recibida. El conocimiento que utiliza es el inicial importado de 'DeclaracionRamas' para conocer las ramas que siguen existiendo (las ramas a las que puede derivar aún si esta no era definitiva), el importado de 'DecisionPrimerCamino' para saber por cual empezaremos y, por último, aquel que va deduciendo sobre la marcha que le ayudará a decidir el camino por el que continuará el razonamiento. En cuanto a conocimiento que se deduce, la respuesta a cada pregunta por parte del estudiante conlleva la creación de un hecho que almacene esta y, a partir de cual sea, disparar alguna regla u otra que continúe deduciendo conocimiento.
- **ModuloIS:** En resumidas cuentas, el objetivo de este módulo es el mismo que el anteriormente mencionado con la única diferencia de plantear diferentes preguntas y posibles bifurcaciones. En cuanto al



conocimiento que utiliza es realmente el mismo y aquel que vaya deduciendo conforme se obtengan resultados y, en cuanto al conocimiento que se deduce, el necesario para decidir si aconsejar esta rama o pasar a preguntar por otra por la que parezca ir más orientado.

- ModuloCSI: Ídem de lo mismo para este módulo con las únicas diferencias del contenido de las preguntas y las reglas.
- ModuloSI: Similar al resto de módulos de preguntas sobre una rama concreta.
- ModuloTI: Misma estructura y similar contenido que las cuatro anteriores.
- VolvemosAPrimera: El objetivo de esta pasa por crear un consejo basado en la primera rama que se decidió explorar ya que, aunque posiblemente inmaduro, existía un criterio por el que se seleccionó como factible y el haber llegado hasta este módulo solamente ha podido ser resultado de haber explorado las cinco ramas sin encontrar una con la que concuerde del todo. Por ello, como no tenemos evidencias que nos conduzcan a ninguna mención en especial, vamos a recomendar la única sobre la que hemos tenido sospechas fundamentadas. El conocimiento que se utiliza es el hecho que almacena desde el principio del programa la decisión sobre qué rama explorar primero, importado desde el módulo 'DecisionPrimerCamino' y el hecho de Consejo creado en el deffacts de 'asignaturas' dentro del módulo 'PreguntasPrevias'. Este último será útil para comprobar si realmente no existe ningún consejo y tenemos que crear uno sobre la primera decisión. El único conocimiento que se puede deducir de aquí es el hecho (Consejo ?rama ?explicacion ?experto) antes de pasar al módulo final de decisión.
- Decision: Debería ser el módulo al que se llegue en último lugar cuando tengamos un consejo definido, ya sea por concordancia con alguna rama o por haber vuelto a la primera original. El objetivo de este módulo no es otro que mostrar por pantalla la decisión adoptada por el sistema al terminar todas las preguntas, además de reconducirte al inicio de este con un (reset) que elimine todos los hechos creados durante la ejecución y permita empezar de nuevo. El conocimiento que utiliza puede venir importado desde cualquiera de los módulos de rama (ModuloTI, ModuloIC, ModuloIS, ModuloSI o ModuloCSI) o del módulo 'VolvemosAPrimera' y no deduce conocimiento alguno porque termina en él.

- Hechos utilizados por cada módulo, deducidos y reglas

- DeclaracionRamas:

En este módulo más que usar hechos los define para los módulos que vienen a continuación. Tampoco deduce conocimiento ni contiene reglas de ningún tipo.

- PreguntasPrevias:

Este módulo utiliza los hechos que se definen en él mismo relacionados con las asignaturas que se van a mostrar por pantalla para que el usuario seleccione su favorita de cada curso. También utiliza reglas creadas localmente para activar las

preguntas o comprobaciones pertinentes en cada momento. Las reglas que deduce son tanto las creadas para el control de las preguntas como las relacionadas con el almacenamiento de las respuestas del estudiante, siendo estas últimas las relevantes para el resto de módulos ya que de ellas dependen la mayoría de las decisiones, como la primera rama por la que preguntar, entre otras. Las reglas contenidas en el módulo son las que muestran las asignaturas por pantalla, realizan la pregunta, reciben la respuesta del usuario, comprueba la validez de estos datos introducidos, preguntan por la media y preguntan por el esfuerzo de este en la carrera.

- **DecisionPrimerCamino:**  
Los hechos que necesita utilizar este módulo son los heredados del módulo 'PreguntasPrevias', ya que contienen la información necesaria para tomar la decisión de la rama por la que empezaremos preguntando. Uno de los hechos que deduce son (empezar\_por ?rama) que contendrá la rama por la que vamos a preguntar en este momento. Cuando dejemos de preguntar por esta rama, este hecho se eliminará y se creará otro del tipo (empezar\_por ?nueva\_rama), aunque esto ocurrirá más adelante. El otro hecho que deduce es (primero\_fue ?rama) que sí que será inalterable a lo largo de todo el proceso y únicamente será útil a la hora de tomar una decisión final en el caso de no habernos decantado por ninguna rama concreta. Las reglas de este módulo realizan dos tipos de comprobaciones: si las asignaturas elegidas son de la misma rama decidirá esa como la inicial sin importar lo demás y si las asignaturas son de ramas diferentes se basa en la media. Están estratégicamente ordenadas para que priorice escoger aquellas ramas con mayor media siempre que la media del alumno acompañe.
- **ModuloControlDeBloque:**  
Este módulo utiliza los hechos creados en el módulo de 'DecisionPrimerCamino' obtenido a través de una importación. No deduce ningún hecho porque solamente redirige el flujo del programa y las reglas dentro del módulo son aquellas que toman estos hechos importados como condiciones para introducir un módulo de destino en pila u otro.
- **ModuloIC, ModuloIS, ModuloCSI, ModuloSI y ModuloTI:**  
Todos estos módulos utilizan hechos heredados tanto de 'DeclaracionRamas' para saber qué ramas no hemos recorrido aún (y, por tanto, podemos rebotar hacia ellas), importadas de 'DecisionPrimerCamino' para activar las acciones de un módulo o no con el hecho (empezar\_por ?rama) y las que se van creando en el interior del módulo que provocan la activación de algunas preguntas u otras, la generación de un consejo o el rebote a otra rama. Los hechos que se inducen son a raíz de las respuestas del usuario y estos pueden ser usados localmente para el control de las preguntas o exportados a módulos finales de decisión para la generación de consejos o finalización del sistema. Las reglas son extensas y variadas con el fin de tocar todas las respuestas posibles por parte del estudiante, cumpliendo el papel de preguntar, aconsejar y redirigir a otro módulo.
- **VolvemosAPrimera:**  
Los hechos necesarios para el funcionamiento de este módulo proceden importados desde 'DecisionPrimerCamino' para conocer cual fue la primera rama sobre la que se decidió explorar y desde 'PreguntasPrevias' para conocer si el consejo que existe en el sistema es el original, representando que no se ha llegado

a otro a lo largo del razonamiento o si este ya no existe y ha sido reemplazado por otro, indicando que no necesitamos volver a la original.

Los hechos que deduce son un consejo y las reglas son para crear un consejo y para ir a 'PreguntasPrevias' y empezar el funcionamiento del sistema. Esto se debe a que, por temas de llamadas a módulos, nos conviene que este módulo sea el primero que se introduzca en pila y con ello el último al que se acceda.

- **Decision:**

Los hechos de este módulo proceden de cualquiera de las últimas seis opciones mencionadas, ya que son aquellas que exportan consejos si se dan. No se deduce ningún hecho por motivo de ser el último que se ejecuta y la regla de su interior tiene la única función de mostrar por pantalla la conclusión a la que llega el sistema, su justificación y el experto que llevó a ello.

Si nos referimos al apartado B, es un Sistema Basado en el Conocimiento que aconseja al alumno en la elección de entre dos asignaturas con las que este tiene dudas. Además de aconsejarle una sobre la otra, debe añadir una explicación que justifique esa decisión. Este sistema incluye un tipo de razonamiento con incertidumbre, más concretamente el de factores de certeza.

Al igual que para el apartado A, se ha perseguido una cómoda gestión de la entrada y salida de datos, una conclusión precisa, un proceso natural de interacción, una modularización simple pero limpia y una explicación que concuerde perfectamente con lo preguntado.

- Variables de entrada

La entrada de datos se realiza por la interfaz de CLIPSIDE con la forma de respuesta a las preguntas que realiza el sistema. A nivel código, esta interacción es posible por el comando (read). Las variables de entrada para este apartado son respuestas por parte del alumno sobre el contenido y herramientas de ambas asignaturas seleccionadas como duda para que el sistema calcule con qué certeza cada rama es la correcta y pueda ofrecer una solución con criterio.

La representación de estas entradas en el sistema consiste en la creación de un hecho que almacene la respuesta. Este hecho debe ser representativo y descriptivo, intentando minimizar la cantidad de hechos necesarios para el funcionamiento del sistema.

- Variables de salida

En cuanto a las salidas del sistema, estas son las dos asignaturas seleccionadas junto a la certeza de que el alumno deba coger cada una, la selección de la mejor hipótesis y la justificación que lo respalde. De nuevo, su representación es un hecho creado a partir de la información introducida, que contiene toda la información necesaria y que será expuesta por la interfaz con la orden (printout). Además, durante el proceso de interacción con el usuario, también muestra preguntas como salidas del sistema, pero no al nivel de la respuesta final.

- Conocimiento del sistema

Inicialmente el sistema parte con un deffacts llamado 'Preguntas' que contiene cuatro preguntas por cada asignatura de las que vamos a darle a elegir con la estructura (Pregunta[1|2|3|4] [NOMBRE\_ASIGNATURA] [TEXTO\_PREGUNTA]). Al tener una base de 10 asignaturas, contamos con 40 hechos como este de los que almacena las preguntas de cada rama. Además, dentro del mismo deffacts, pero rompiendo con el

patrón del resto, contamos con un hecho de la forma (mayor\_hipotesis a 0) que representa que nuestra mayor hipótesis tiene una certeza de 0 y un nombre de asignatura recomendada 'a' inicialmente. Esta es la forma de inicializar un valor que más tarde actualizaremos y recorreremos en busca del mayor.

Por otra parte, cuenta con otra estructura de hechos llamada 'asignaturas' que almacena todas las asignaturas que ofreceremos al alumno por pantalla de entre las que tendrá que elegir dos que le generen dudas. Estos hechos son tal que (asignatura [NOMBRE\_ASIGNATURA]).

- Módulos desarrollados

La modularización llevada a cabo es aquella que hace uso de estructuras 'defmodule'. Contábamos con la alternativa de modularización con hechos de control o la asignación de la misma prioridad a todas las reglas de una fase, siendo esta mayor que las que correspondan con una fase posterior, pero por temas de facilidad y documentación la elección ha sido utilizar la herramienta que proporciona el propio CLIPS para definir módulos.

○ *Estructura en módulos.* Módulos definidos en el sistema:

- Modulo\_asignaturas
- Primeras\_preguntas

○ *Descripción de los módulos.*

- Modulo\_asignaturas: Este módulo inicial tiene únicamente el objetivo de definir los hechos que contienen las preguntas por cada asignatura con las que se puede tener dudas. Para ello contiene un deffacts que contiene cuatro preguntas por cada una de las 10 asignaturas con el formato (Pregunta1|2|3|4 [NOMBRE\_ASIGNATURA] [PREGUNTA]). Como extra, se ha aprovechado la estructura de definición de hechos iniciales para introducir en el conocimiento de partida el hecho (mayor\_hipotesis a 0). Este será sustituido siempre que hallemos otra asignatura que tenga más certeza que esta y, en caso en el que no ocurra, el hecho disparará una regla que informe de que la combinación introducida no ayuda al sistema a decidir una de las dos asignaturas. El módulo no hace uso de ningún conocimiento ni deduce este como tal, simplemente lo declara y exporta para hacerlo accesible desde diferentes puntos del sistema.
- Primeras\_preguntas: Por otra parte, este otro módulo tiene el objetivo de realizar el grueso del apartado, realizando preguntas y utilizando las respuestas para crear factores de certeza asociados a un valor numérico que llevarán al sistema a decidirse por una de las asignaturas como más acertada para el alumno. El proceso de razonamiento se especifica en un apartado dedicado únicamente a ello más adelante. El conocimiento del que hace uso es aquel que introduce el estudiante como respuesta a las preguntas que este mismo módulo plantea y el que induce se puede ver en dos grupos: hechos intermediarios que activan y desactivan preguntas y fases y, por otra parte, el conocimiento definitivo asociado a la decisión final como el conjunto de todas las hipótesis, su certeza, la mejor de ellas y las explicaciones de cómo llegó hasta él.

- Hechos utilizados por cada módulo, deducidos y reglas

- **Modulo\_asignaturas:**

En este módulo no se hace uso de ningún hecho ni se llega a deducir nada, únicamente existe la declaración de ciertos hechos. No hay reglas dentro debido a que el fin del módulo es ser utilizado como almacén de información al que recurrir desde donde haga falta cuando sea necesario.

- **Primeras\_preguntas:**

Aquí los hechos utilizados se pueden ver en diferentes grupos. Para empezar, se usan hechos importados del módulo 'Modulo\_asignaturas' con dos objetivos. El primero de ellos es para realizar las preguntas de cada asignatura, ya que estas tienen que estar relacionadas con ellas y el segundo es en la búsqueda de hipótesis o de la mejor de ellas. Para el primer objetivo se hace uso de las reglas que siguen la forma de (Pregunta [ASIGNATURA] [PREGUNTA]) y para el segundo el hecho (mayor\_hipotesis a 0) que será sobrescrito en cuanto se cree una hipótesis con certeza positiva. Otro grupo de hechos que se van a utilizar son los que siguen la forma (asignatura [NOMBRE\_ASIGNATURA]) definidos como conocimiento inicial dentro de un defacts al inicio del módulo que contiene las 10 asignaturas que serán mostradas por pantalla como menú de elección. Otros hechos utilizados serán aquellos intermedios que se creen a partir de las respuestas del usuario y servirán como condiciones de activación de preguntas y reglas lógicas. Estos hechos también pueden ser considerados como aquellos que se deducen, pues nacen a partir de la información proporcionada por el estudiante. Por otra parte, el conocimiento deducido que más peso tiene en la decisión es el que se genera una vez el sistema tiene las certezas definitivas y dispararán el final del programa.

En cuanto a las reglas dentro del módulo están las que realizan las preguntas iniciales, las que comprueban la validez de las respuestas, las que lanzan las preguntas relacionadas con cada asignatura, una por cada regla lógica individual relacionada a cada pregunta, cuatro reglas que utilizan la combinación para ser expuesto como ejemplo, las que seleccionan la mejor hipótesis entre todas y las que muestran la decisión + justificación por pantalla.

Como mención especial se explicará el módulo inicial que es independiente a cualquiera de los apartados anteriores, pero tiene una labor imprescindible. Este lleva una orden de (declare (auto-focus TRUE)) que indica que es el primer módulo que se va a cargar en la pila. Este es el primer paso en el funcionamiento del sistema, elegir si queremos asesoramiento sobre ramas o asignaturas o, en su defecto, sobre nada y parar la ejecución. Se realiza una pregunta al usuario que será almacenada y determinará que conjunto de módulos se activan, si los de las asignaturas o los de la rama. Por este módulo no se vuelve a pasar hasta que se termine en alguno de los dos apartados y se despliegue la opción de volver a empezar en alguno de los dos, siendo aquí donde volvemos a tener la opción de detener el uso del sistema.

## Como se justifica el razonamiento

El razonamiento se ve guiado por las respuestas del estudiante relacionadas con cada pregunta. El proceso interno es el siguiente:

- El usuario elige dos asignaturas que serán entre las que decidirá el sistema (siguiendo las restricciones de ser asignaturas para las que tenemos preguntas y que la segunda sea diferente a la primera, ya que si no estas no se cumplen, el sistema pide los datos hasta recibir unos válidos)
- Cuando el sistema tiene estas dos asignaturas almacenadas, accede a las cuatro preguntas de cada asignatura y las realiza al usuario almacenando (y comprobando) las respuestas que este introduce.
- Estas respuestas se almacenan en hechos como el siguiente: (respuesta respuesta[1|2|3|4] [NOMBRE\_ASIGNATURA] si|no) donde la primera palabra 'respuesta' es fija y sirve para identificar el tipo de hecho que es, la segunda palabra puede ser respuesta1, ..., respuesta4 dependiendo de si se corresponde con la respuesta de la primera, segunda, tercera o cuarta pregunta de esa asignatura; el tercer elemento de esa regla contiene el nombre de una de las asignaturas elegidas por el estudiante y por último un 'si' o 'no' dependiendo de la respuesta introducida (por eso se comprueba que lo que se introduzca sea correcto, aceptar respuestas inválidas llevaría a la no activación de reglas imprescindibles para el correcto funcionamiento)
- En cuanto se comprueba la validez de los hechos anteriormente descritos, una regla se encarga de transformarlos en hechos como (FactorCerteza respuesta[1|2|3|4] [NOMBRE\_ASIGNATURA] si|no 1), que servirán como condiciones para las reglas lógicas que asignan certezas. El 1 del final representa la certeza de este hecho, ya que al ser una respuesta del usuario, es la máxima.
- Cada pregunta tiene asociada una regla independiente que, de dispararse, generaría un hecho con el nombre de la asignatura y la certeza. Para verlo más claro, un ejemplo sacado del código sería:

```
(defrule R2
```

```
  (FactorCerteza respuesta2 IA si ?f1)
```

```
  (test (> ?f1 0))
```

```
=>
```

```
  (assert (FactorCerteza IA si 0.4))
```

```
  (bind ?expl (str-cat " POSITIVO: El algoritmo A* es la base de la segunda practica, por lo que conocerlo hara que disfrutes la asignatura"))
```

```
  (assert (explicacion IA ?expl))
```

```
)
```

Y representa que si has seleccionado IA como una de las asignaturas entre las que decidir y además has respondido 'si' a su segunda pregunta (ahora mismo irrelevante, pero en este caso es "Has oído hablar antes del algoritmo A\*?"), se crea un FactorCerteza con certeza de 0.4 y se crea una explicación de por qué esa respuesta ha

influido en la decisión de esta asignatura, por si termina siendo la seleccionada, que sepamos el camino que le ha llevado a la decisión.

Siguiendo este procedimiento se pueden terminar creando varios FactoresCerteza de la misma asignatura, pero con diferentes valores de certeza. Para corregir esto, existe una función proporcionada en el ejemplo de PRADO que combina las certezas siguiendo una fórmula concreta en la que no entraremos. Finalmente, la combinación de estas certezas (positivas y negativas) resultará en un valor que se corresponda proporcionalmente con lo introducido.

- Además de cada regla lógica individual, se han implementado 4 reglas que hacen uso de otra función proporcionada en PRADO que combina condiciones que al darse a la vez tienen un efecto diferente que el que tienen en la activación independiente de cada condición de las que se da. Por ejemplo, aunque sea una respuesta correspondiente a la asignatura de estadística, si indicas que no eres bueno memorizando fórmulas y también has dicho que no se te da bien integrar, afecta negativamente a la elección de cálculo por encima de lo que pueda afectar a estadísticas. La forma de disparar y probar estos casos será especificada en el siguiente apartado.
- Un error que puede contener el sistema es que, al haber una regla por cada pregunta, solo se activará si la respuesta coincide con la condición que debe darse, por lo que, si casualmente introdujésemos la combinación exacta de ocho respuestas opuestas a las utilizadas, no se dispararía ninguna regla a pesar de tener información suficiente para un veredicto. Es por esto por lo que se han implementado 1 regla extra opuesta a una aleatoria de cada asignatura, haciendo que para cada asignatura, al menos una pregunta, independientemente de la respuesta que tenga, va a disparar una regla que nos acerque o aleje de esta asignatura.
- Una vez se obtiene la certeza definitiva de cada asignatura, habrá almacenadas explicaciones positivas y negativas de cada asignatura justificando el por qué de esa certeza final, pero el sistema está implementado para que además de mostrar la certeza de ambas asignaturas, seleccione la que mayor valor tiene y muestre únicamente su explicación. Esto se hace para cumplir con el punto del enunciado de “El sistema debe ir comunicando al usuario el razonamiento que ha hecho y que sea **relevante**”.

## Manual de uso del sistema

Para el uso del sistema explicaremos el proceso por el que se pone en marcha, las respuestas que se pueden ingresar y aquellas que se deben ingresar para llegar a los casos de ejemplo que mostraremos a continuación.

- *Ponerlo en marcha*

Dentro del entorno de CLIPSIDE debemos poner el programa en marcha introduciendo los siguientes comandos:

- (clear)
- (load “ruta\_absoluta/nombre\_archivo.clp”)
- (reset)
- (run)

```
Dialog Window
CLIPS (6.31 6/12/19)
CLIPS> (clear)
CLIPS> (load "C:/Users/xaviv/Desktop/JAVIER/UNIVERSIDAD/CURSO 20.21/CURSO/2CUATRI/INGENIERIA DEL CONOCIMIENTO
/PRACTICA/PRACTICA2/APARTADO 3/JAVI.clp")
```

```
Dialog Window
Defining defrule: R302 +j+j
Defining defrule: R31 +j+j
Defining defrule: R32 +j+j
Defining defrule: R33 +j+j
Defining defrule: R34 +j+j
Defining defrule: R342 +j+j
Defining defrule: R35 +j+j
Defining defrule: R36 +j+j
Defining defrule: R37 +j+j
Defining defrule: R38 +j+j
Defining defrule: R382 +j+j
Defining defrule: R39 +j+j
Defining defrule: R40 +j+j
Defining deffunction: combinacion
Defining defrule: combinar +j+j+j
Defining defrule: combinar_signo +j+j+j
Defining defrule: factorcerteza_problema +j+j
Defining defrule: factorcerteza_problema_mayor +j+j+j
Defining defrule: factorcerteza_problema_igual +j+j+j+j
Defining defrule: muestra_todos +j+j
Defining defrule: muestra_mayor +j+j+j
Defining defrule: muestra_motivos =j=j+j+j
Defining defrule: muestra_empate +j+j+j+j+j+j
Defining defrule: ningun_problema +j+j
Defining defrule: reinicio +j+j
TRUE
CLIPS> (reset)
CLIPS> (run)
```

- *Dentro del programa cuando lo ponemos a ejecutar:*

Una vez el programa esté ejecutando, el funcionamiento es intuitivo y podemos limitarnos a seguir los pasos que el sistema nos indique y solicite. Aún así, en líneas generales lo primero que haremos es elegir si queremos la parte del sistema de 'ramas' para acceder al apartado A, 'asignaturas' para el apartado B y 'nada' para detener el programa.

Según la parte que escojamos, responderemos unas preguntas u otras, pero el dominio de las respuestas es especificado en cada momento y si introducimos algo que no sea válido, habrá que repetirlo hasta que lo sea, por lo que es difícil no saber cómo hacer avanzar el sistema.

A continuación, voy a mostrar un ejemplo de ejecución aleatorio en el que empezamos por las ramas, luego por las asignaturas y probamos a parar el sistema, lo cual será útil para conocer cómo funciona y cómo son los datos que se introducen.



=====

Sobre que quiere asesoramiento? Sobre 'ramas' o sobre 'asignaturas'? (Para terminar introduzca 'nada')

ramas

Cual de las siguientes es tu asignatura favorita de primero?:

Ninguna  
LMD  
FFT  
FS  
ES  
ALEM  
CAL  
TOC  
IES  
FP  
MP

MP

Cual de las siguientes es tu asignatura favorita de segundo?:

Ninguna  
ED  
AC  
EC  
SCD  
FBD  
ALG  
SO  
PDOO  
FIS  
IA

FIS

Cual es tu nota media actual?: (SUFICIENTE (5), BIEN (6), NOTABLE (7-8), SOBRESALIENTE (9-10))

SUFICIENTE

Como consideras tu esfuerzo en la carrera? (ALTO/MEDIO/BAJO)

BAJO

Muy bien, vamos a elegir la rama con la que mas concuerdas...

Te gustan los servidores? (SI/NO/NO SE)

SI

<

Muy bien, vamos a elegir la rama con la que mas concuerdas...

Te gustan los servidores? (SI/NO/NO SE)

SI

Te gustan los temarios sobre redes? (SI/NO/NO SE)

NO

Te gusta tecnologia web? (SI/NO/NO SE)

NO

Te gustan la arquitectura del sistema? (SI/NO/NO SE)

SI

Te gusta el hardware digital? (SI/NO/NO SE)

NO

Te gustan los microprocesadores y los elementos fisicos del sistema? (SI/NO/NO SE)

NO SE

Te gustan las matematicas? (SI/NO/NO SE)

NO

Te gusta la programacion? (SI/NO/NO SE)

SI

Te gusta la Inteligencia Artificial? (SI/NO/NO SE)

NO

Te gusta mas el hardware que el software? (SI/NO/NO SE)

SI

Te gusta la programacion web? (SI/NO/NO SE)

SI

Te gusta hacer sistemas de informacion para empresas? (SI/NO/NO SE)

NO

Te gustan las bases de datos? (SI/NO/NO SE)

SI

JavierRamirez te recomienda la rama Sistemas\_de\_Informacion porque Te\_gusta\_mayormente\_el\_contenido\_de\_la\_rama\_y\_no\_te\_esfuerzas\_mucho

=====

Sobre que quiere asesoramiento? Sobre 'ramas' o sobre 'asignaturas'? (Para terminar introduzca 'nada')

<

```

-----
Sobre que quiere asesoramiento? Sobre 'ramas' o sobre 'asignaturas'? (Para terminar introduzca 'nada')
asignaturas
Sobre cual de las siguientes asignaturas tienes dudas?:
    IA
    PDOO
    FBD
    EC
    AC
    ES
    CAL
    TOC
    IES
    MP
CAL
Cual es la otra asignatura con la que dudas?:
    IA
    PDOO
    FBD
    EC
    AC
    ES
    TOC
    IES
    MP
MP
Has utilizado alguna vez la IDE de NetBeans? (si/no)
si
Manejas el concepto de clases en programacion? (si/no)
no
Tienes un nivel medio por lo menos en C++? (si/no)
no
Sabes como funcionan los punteros? (si/no)
si
Conoces el polinomio de Taylor? (si/no)
si
Conoces la herramienta de Maxima? (si/no)
si
Eres bueno intengrando y derivando? (si/no)
<

Eres bueno intengrando y derivando? (si/no)
no
Sabes lo que son los numeros imaginarios y como usarlos? (si/no)
si
----- CERTEZA DE LA ASIGNATURA CAL 0.98
----- CERTEZA DE LA ASIGNATURA MP 0.6666666666666667
La mayor hipotesis es CAL y los motivos son:
POSITIVO: Las practicas se realizan usando la herramienta de Maxima, por lo que tener conocimientos previos te va a poner la practica en bandeja
POSITIVO: Para el segundo parcial es crucial saber manejar el polinomio de Taylor, por lo que conocerlo previamente ayuda bastante
-----
-----
-----
Sobre que quiere asesoramiento? Sobre 'ramas' o sobre 'asignaturas'? (Para terminar introduzca 'nada')
nada
Perfecto, muchas gracias!
CLIPS> █
<

```

Las siguientes capturas son las muestras del comportamiento del sistema en los casos en el que respondemos erróneamente:

```

Sobre que quiere asesoramiento? Sobre 'ramas' o sobre 'asignaturas'? (Para terminar introduzca 'nada')
iifiejdiejdiiede
Lo siento, la opcion introducida no es correcta, escriba 'asignaturas', 'ramas' o 'nada'
jiedajf
Lo siento, la opcion introducida no es correcta, escriba 'asignaturas', 'ramas' o 'nada'
ajdiae
Lo siento, la opcion introducida no es correcta, escriba 'asignaturas', 'ramas' o 'nada'
asignaturas
Sobre cual de las siguientes asignaturas tienes dudas?:
IA
PDOO
FED
EC
AC
ES
CAL
TOC
IES
MP
nedade
nedade no es una opcion correcta, introduce otra
IA
PDOO
FED
EC
AC
ES
CAL
TOC
IES
MP
CAL
Cual es la otra asignatura con la que dudas?:
IA
PDOO
FED
EC
AC
ES
TOC
IES
MP
CAL
No puede seleccionar la misma que eligio como primera opcion, introduce otra
Cual es la otra asignatura con la que dudas?:
IA
PDOO
FED
EC
AC
ES
TOC
IES
MP

```

```

Has utilizado alguna vez la IDE de NetBeans? (si/no)
nfsirs
La respuesta no es correcta, tienes que introducir si o no
no
Manejas el concepto de clases en programacion? (si/no)
si
Tienes un nivel medio por lo menos en C++? (si/no)
si
Sabes como funcionan los punteros? (si/no)
no
Conoces el polinomio de Taylor? (si/no)
diaed
La respuesta no es correcta, tienes que introducir si o no
no
Conoces la herramienta de Maxima? (si/no)
si
Eres bueno intengrando y derivando? (si/no)
no
Sabes lo que son los numeros imaginarios y como usarlos? (si/no)
si
----- CERTEZA DE LA ASIGNATURA CAL 0.857142857142857
----- CERTEZA DE LA ASIGNATURA MP 0.6

La mayor hipotesis es CAL y los motivos son:
POSITIVO: Las practicas se realizan usando la herramienta de Maxima, por lo que tener conocimientos previos te va a poner la practica en bandeja
NEGATIVO: Para el segundo parcial es crucial saber manejar el polinomio de Taylor, por lo que no conocerlo no ayuda precisamente

```

Y por último los pasos a seguir para disparar reglas combinadas. A pesar de que hay varias, la que vamos a provocar es la que combina EC y AC:

```
(defrule COMBINADA2
```

```
(FactorCerteza respuesta4 EC si ?f1)
```

```
(FactorCerteza respuesta1 AC si ?f2)
```

```
(test (and (> ?f1 0) (> ?f2 0)))
```

```
=>
```

```
(assert (FactorCerteza EC si (encadenado (* ?f1 ?f2) 0.7)))
```

```
(bind ?expl (str-cat " POSITIVO: Si ademas te interesa en general el interior del pc tambien te gustara la asignatura de EC en gran medida"))
```

```
(assert (explicacion EC ?expl))
```

```
)
```

Vemos que si se ha respondido que sí a la pregunta 4 de EC (“Te genera interes el interior de un procesador y sus elementos?”) y que si a la pregunta 1 de AC (“Te gusta la parte fisica del ordenador?”) afecta positivamente a que te guste EC, ya que una cosa puede implicar la otra.

Se muestra en las siguientes imágenes el proceso para que esta se dispare.

```
-----
Sobre que quiere asesoramiento? Sobre 'ramas' o sobre 'asignaturas'? (Para terminar introduzca 'nada')
asignaturas
Sobre cual de las siguientes asignaturas tienes dudas?:
    IA
    PDOO
    FED
    EC
    AC
    ES
    CAL
    TOC
    IES
    MP
EC
Cual es la otra asignatura con la que dudas?:
    IA
    PDOO
    FED
    AC
    ES
    CAL
    TOC
    IES
    MP
AC
Te gusta la parte fisica del ordenador? (si/no)
si
Conoces la ley de Andahl? (si/no)
no
Sabes la diferencia entre el protocolo MSI y MESI? (si/no)
si
Has trabajado alguna vez con OpenMP? (si/no)
no
Has trabajado en ensamblador alguna vez? (si/no)
no
Has hecho uso de Arduinos alguna vez? (si/no)
no
Tienes buen manejo del lenguaje C? (si/no)
si
Te genera interes el interior de un procesador y sus elementos? (si/no)
si
----- CERTEZA DE LA ASIGNATURA EC 0.64
----- CERTEZA DE LA ASIGNATURA AC 0.857142857142857
La mayor hipotesis es AC y los motivos son:
NEGATIVO: OpenMP es una herramienta que se utiliza para la parte de los cerrojos y haber trabajado con ella habria sido una gran ventaja
POSITIVO: Conocer los protocolos MSI y MESI te ahorra las partes mas complejas de las relaciones de ejercicios, por lo que no habria que desaprovechar
NEGATIVO: La ley de Andahl es uno de los principios que se estudian en teoria y no conocerla influye de forma negativa pero no excesivamente
POSITIVO: Una parte de la teoria de la asignatura esta basada en componentes del ordenador
=====
=====
=====
=====
Sobre que quiere asesoramiento? Sobre 'ramas' o sobre 'asignaturas'? (Para terminar introduzca 'nada')
■
```