

Construcción de entornos de aprendizaje seguro para la enseñanza de programación

Building safe learning environments in programming education

Jorge Calvillo Arbizu

ORCID: <https://orcid.org/0000-0003-1277-3310>

Universidad de Sevilla

Departamento de Ingeniería Telemática

jcalvillo@us.es

DOI: <http://dx.doi.org/10.12795/9788447222865.136>

Pp.: 2383-2399



Resumen

El aprendizaje de un lenguaje de programación, sobre todo cuando no se poseen experiencias previas, es una tarea harto difícil y requiere de una ajustada combinación de teoría y práctica con el fin de adquirir las destrezas y habilidades necesarias para resolver problemas reales. La programación se sustenta sobre un proceso donde el pensamiento crítico y el interés por la resolución de problemas son aspectos esenciales y donde equivocarse es consustancial al proceso de aprendizaje. Por ello, consideramos que el aprendizaje en programación debe darse en un entorno seguro donde los estudiantes se atrevan a exponer su opinión sin miedo a equivocarse. Este trabajo presenta el diseño, desarrollo y evaluación de un Ciclo de Mejora en el Aula (CIMA) en la asignatura Fundamentos de Programación I del Grado en Ingeniería de las Tecnologías de Telecomunicación de la Universidad de Sevilla en la que se pretende formar a los estudiantes en la programación mediante el lenguaje C. La metodología de innovación se basa en la construcción de un entorno de aprendizaje seguro y en ejercicios que trabajan los estudiantes para potenciar su zona de desarrollo próximo con el fin de facilitar el aprendizaje de contenidos conceptuales, procedimentales y actitudinales de resolución de problemas.

Palabras clave: Fundamentos de programación I, grado en ingeniería de las tecnologías de telecomunicación, docencia universitaria, desarrollo profesional docente.

Abstract

Learning a programming language, especially without previous experience, is a very difficult task that requires a tight combination of theory and practice in order to acquire the skills and abilities necessary to solve real problems. Programming is based on a process where critical thinking and an interest in problem solving are essential aspects and where making mistakes is inherent to the learning process. For this reason, we consider that programming learning should take place in a safe environment where students dare to express their opinion without fear of being wrong. This work presents the design, development and evaluation of a cycle of teaching innovation in the subject Fundamentals of Programming I of the Degree in Engineering of Telecommunication Technologies of the University of Seville in which it is intended to train students in programming with the language C. The innovation methodology is based on the construction of a safe learning environment and on exercises that students work on to enhance their zone of proximal development in order to facilitate the learning of conceptual, procedural and attitudinal problem-solving content.

Keywords: Fundamentals of programming I, degree in engineering of telecommunication technologies, university teaching, teaching professional development.



Introducción

Descripción del contexto

La experimentación docente universitaria se ha llevado a cabo para la asignatura obligatoria *Fundamentos de Programación I* que se imparte en el primer cuatrimestre del primer curso del Grado en Ingeniería de las Tecnologías de Telecomunicación ofertado por la Escuela Técnica Superior de Ingeniería. El aula en la que se desarrolla el curso es estándar con los medios audiovisuales tradicionales (ordenador, proyector, etc.).

El Ciclo de Mejora en el Aula (CIMA) (Delor, Hamed y otros, 2020) se destina a un grupo de alrededor de 60 alumnos de los cuales aproximadamente el 10 % son repetidores mientras que el resto son alumnos recién ingresados en la enseñanza universitaria. Este hecho es de especial importancia puesto que la mayoría de los alumnos cuentan con limitados o nulos conocimientos en la materia de la asignatura.

Conexión con el proceso previo

Aunque los CIMA's realizados anteriormente fueron aplicados a una asignatura diferente, algunas de las conclusiones obtenidas en dichas experimentaciones son de aplicación aquí (Calvillo-Arbizu, 2018). En el CIMA inmediatamente anterior se desarrolló una experiencia basada en gamificación que fue acogida con gran éxito entre los alumnos. Este hecho permite concluir que la captación del interés del alumnado, haciéndolo partícipe de la enseñanza de manera explícita, es un elemento activador del proceso de aprendizaje de gran valor. Por otro lado, otra conclusión clave del proceso previo es la necesidad de priorizar los contenidos, haciendo hincapié en aquellos que suponen mayor dificultad en el aprendizaje y facilitando a su vez que los estudiantes trabajen de manera autónoma fuera del aula.

Igualmente es necesario conocer el punto de partida de los modelos mentales de los estudiantes pues estos determinarán la facilidad o dificultad del aprendizaje de nuestra materia. Por último, la utilización de un diario para evaluar la ejecución de las clases y poder revisarlas y analizarlas con posterioridad es un método de gran valor para mantener una mejora progresiva y continuada.

El CIMA que se presenta a continuación está construido sobre los principios anteriores y se caracteriza por:

- *Un mapa de contenidos de la asignatura completa que diferencia contenidos conceptuales, procedimentales y actitudinales.*



- *Un modelo metodológico posible*, que es el que vamos a implementar en esta experimentación, que se encuentra entre el modelo metodológico actual y el ideal.
- *La secuencia de actividades* que se desarrollará en las ocho sesiones que componen el CIMA. El seguimiento de la evolución de los modelos mentales de los estudiantes se llevará a cabo comparando los resultados de un cuestionario completado por los alumnos antes y después de la aplicación del ciclo de mejora. Las observaciones recogidas en el diario del profesor y las encuestas de opinión a los alumnos servirán para evaluar la ejecución y adecuación del CIMA.

Diseño previo del CIMA

Mapa o red de contenidos o problemas/casos

Una de las claves de esta experimentación ha sido el mapa de contenidos que tiene un doble propósito. Por un lado, nos ha servido para reflexionar e identificar los contenidos conceptuales, procedimentales y actitudinales (Bain, 2007) que deseábamos que el alumnado hubiera asimilado al finalizar la asignatura Fundamentos de Programación I. El mapa actúa como red, relacionando los conceptos entre sí y proporcionando una visión integradora y coherente de los mismos. Por ello, el segundo uso del mapa ha sido el de facilitar un vehículo de comprensión de la asignatura, y como tal lo hemos utilizado en las sesiones con los alumnos.

El mapa de contenidos se presenta en la figura 1 y, como hemos indicado, cubre contenidos conceptuales, procedimentales y actitudinales. Los contenidos conceptuales de la asignatura se han separado en dos bloques (elementos básicos y avanzados) para destacar la relevancia de los conceptos fundamentales que hay que dominar en la asignatura para poder avanzar hacia los más complejos. Aunque el mapa se ha desarrollado para cubrir todos los contenidos de la asignatura, el CIMA se ha desarrollado durante las sesiones dedicadas a los contenidos conceptuales básicos.

Dentro de cada bloque se destaca en negrita aquellos conceptos fundamentales que requieren mayor dedicación, los cuales debíamos priorizar en las sesiones. En algunos casos (por ejemplo, en el tema de estructuras de control) era esencial explicar los conceptos primordiales mientras que el detalle de estos podría solo mencionarse y que fueran los alumnos los que lo completasen con los manuales de la asignatura de forma autónoma.

Además de los contenidos conceptuales ya hemos apuntado que existen contenidos procedimentales y actitudinales. Cuando se trata de realizar una actividad de programación (en el aula o en el ejercicio profesional),



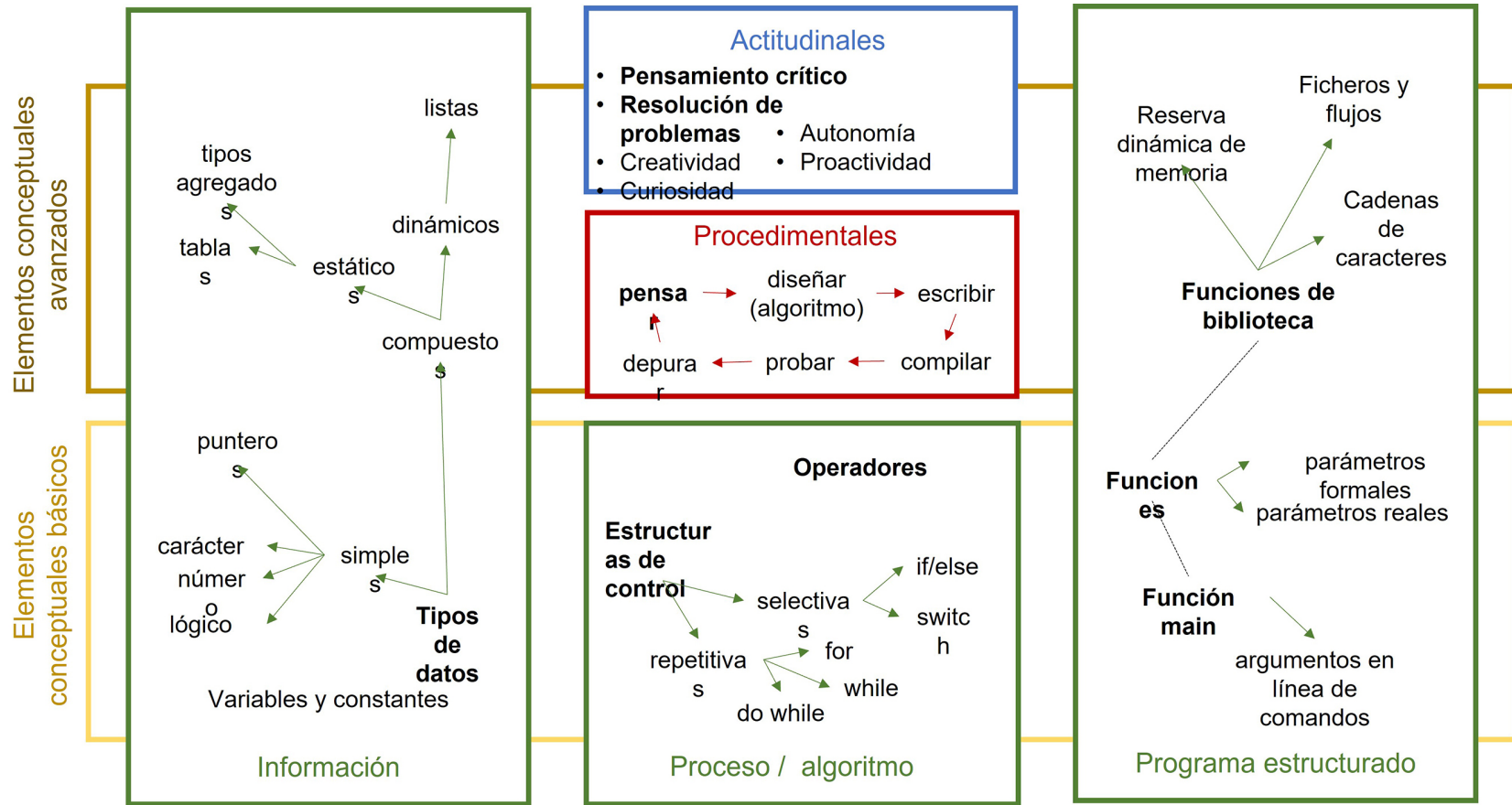


Figura 1. Mapa de contenidos de la asignatura Fundamentos de Programación I.



el proceso más adecuado es el que presentamos en el cuadro rojo de la figura 1. La primera acción siempre debe ser pensar. Pensar en el problema que queremos resolver, identificar los requisitos que debemos satisfacer y las herramientas que tenemos a nuestro alcance. Una vez completada esta primera tarea, el programador pasa a diseñar la solución al problema mediante un algoritmo que bien puede ser en papel, de forma gráfica, por escrito, etc. Solo entonces debemos ir al ordenador a escribir nuestro programa, probarlo y depurar sus errores. Si nuestra solución no se ajusta a lo que estábamos esperando o no resuelve el problema deberemos entrar otra vez en el bucle y repensar el asunto.

Dentro de este proceso, la actividad primordial es la de pensar o razonar, y por eso está resaltada en el mapa de contenidos. Es el contenido procedimental esencial. De cursos anteriores sabemos que algunos alumnos tienden a aprender las diferentes herramientas de que se compone el lenguaje de programación (tipos de datos, estructuras, funciones, etc.), pero no razonan correctamente en cuándo se debe aplicar uno u otro. Lo hacen por intuición, porque recuerdan la solución de problemas similares o completamente al azar. Pero si antes de programar nos dedicamos un tiempo a pensar y razonar, entonces tenemos mayores probabilidades de resolver el problema con éxito. Además, en el futuro y con mucha probabilidad utilizarán lenguajes de programación y herramientas diferentes a las que se presentan en esta asignatura. Lo que nunca va a cambiar es la necesidad de razonar ante el problema, analizar sus características y diseñar la solución.

A continuación, la programación podrá realizarse a través de diferentes lenguajes. Esta clave es esencial para comprender la asignatura y superarla con éxito. La segunda clave procedimental que se necesita detallar desde el principio del curso es que programar es un proceso cíclico; es decir, proponemos una solución, probamos y, si falla, proponemos otra. Por eso, el alumnado debe entender que *el error es consustancial al aprendizaje*, no algo negativo. Un error nos acerca a la solución porque nos permite descartar los elementos más débiles de los diseños o un diseño incorrecto. Programar no es responder a una pregunta donde la respuesta es única. Programar consiste en resolver un problema de la forma más eficiente posible. Pero hay muchas soluciones posibles, y algunas soluciones son mejores que otras. Por eso, en todo este proceso el alumno debe ser proactivo en la resolución del problema, proponiendo soluciones, probándolas, analizando los errores que aparezcan y modificando sus propias propuestas hasta alcanzar la solución óptima.

Finalmente tenemos que destacar también los contenidos actitudinales esenciales, los cuales están íntimamente relacionados con los procedimentales que acabamos de comentar. Las dos actitudes esenciales son *el pensamiento crítico y el deseo por la resolución de problemas*. Ambas



actitudes se combinan en la voluntad por encontrar la mejor solución posible a un problema, que es transversal a todo el Grado y la profesión de ingeniero. Por eso las consideramos esenciales en la formación de nuestro alumnado.

Modelo metodológico posible

Antes de describir el modelo metodológico posible que se ha implementado en el CIMA conviene describir el modelo metodológico de partida, así como el ideal al que aspiramos en el futuro.

El desarrollo de una sesión en el modelo metodológico previo (figura 2) parte de la explicación de contenidos conceptuales (T) que se ilustran a través de ejercicios (E). Algunos de estos ejemplos están resueltos y otros suponen una aplicación de la teoría en la que se solicita la participación del alumnado. Entre las respuestas de los alumnos se debate cuál sería la solución y se resuelven así mismo las dudas que hayan surgido con los ejercicios (D²). La sesión termina con una recapitulación de los contenidos explicados y conclusiones (C).

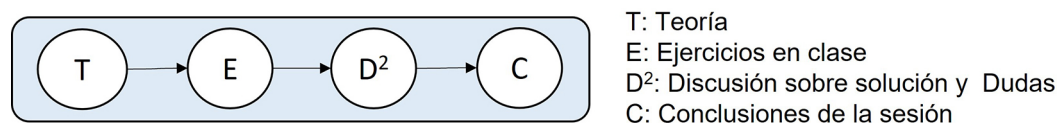


Figura 2. Modelo metodológico actual de Fundamentos de Programación I.

Analizando la metodología actual encontramos que la teoría es la que guiaba el proceso de aprendizaje en la sesión. Sin embargo, dicha teoría no estaba puesta en contexto ni los alumnos entendían qué pregunta o problema venía a resolver. Es decir, era una metodología basada en los contenidos conceptuales que se tienen que impartir en la asignatura, y no se les permitía a los alumnos razonar previamente sobre el problema cuya solución será la teoría. Así, el aprendizaje con esta metodología restringe el proceso a lo que ocurre dentro del aula, ya que es ahí donde se presentan los contenidos y donde el alumno puede seguir el razonamiento del curso. Fuera del aula el alumno solo puede repasar los contenidos ya explicados y esperar a que en sucesivas sesiones se amplíen los conceptos.

Dado que la programación es una materia eminentemente práctica y que existen multitud de herramientas disponibles para su aprendizaje, el modelo metodológico ideal estaría basado en un aprendizaje autónomo por parte del alumno fuera del aula mientras que en las sesiones con el profesor se profundizaría en los obstáculos que hayan encontrado en el



aprendizaje, resolviendo problemas que previamente los alumnos se hayan planteado e intentado resolver, así como animar en el pensamiento crítico y la proactividad de los alumnos.

Así, para el modelo metodológico ideal deberíamos contar con un problema complejo que queramos resolver desde el inicio del curso y que nos sirva de hilo conductor de todos los contenidos conceptuales. El problema debería ser suficientemente interesante para que estimule a los alumnos a resolverlo de forma autónoma para lo cual tendrán que aprender los medios para hacerlo. Ese problema general se descompondría en problemas más pequeños, que nos permitirían adentrarnos paulatinamente en los contenidos de la asignatura, desde los más básicos a los más complejos. El trabajo sería básicamente desarrollado fuera del aula por los alumnos, mientras que en las sesiones regladas se llevarían a cabo solo puntualizaciones y correcciones, se analizaría el aprendizaje de los alumnos, puesta en común de las dificultades, animar en el aprendizaje, reforzar algún contenido conceptual que no haya quedado claro, etc.

Aunque debemos aspirar al modelo ideal, consideramos que aplicarlo conllevaría un salto demasiado grande desde el modelo actual y aún no contamos con garantías reales de que vaya a mejorar el aprendizaje sustancialmente. Por ello, se propuso un modelo metodológico posible que está en un término medio entre el actual y el ideal, y cuyo desempeño y evaluación nos puede permitir conocer si estamos en la dirección correcta.

El modelo metodológico posible se presenta en la figura 3. En un primer lugar, queremos hacer hincapié en la construcción de un entorno de aprendizaje seguro donde todos juntos (el profesor y los alumnos) participamos de un aprendizaje común, donde el error no es negativo sino una clave para el aprendizaje, y donde la realimentación es fundamental para el progreso. Por ello, todas las sesiones comienzan con un tiempo para hablar con los alumnos sobre cómo se encuentran, cómo llevan esta asignatura y el resto (recordemos que los alumnos son recién ingresados en la universidad y a menudo se encuentran perdidos en los primeros meses, desorientados y desilusionados, lo que limita su aprendizaje), así como sus expectativas y miedos (Prev). En las primeras sesiones se anima a la participación preguntando a alumnos aleatoriamente, pero se pretende crear un espacio seguro donde cualquiera pueda hablar libremente. Esta primera etapa de la sesión facilitará la construcción de un entorno amigable sobre el que entraremos ya en los contenidos propios de la asignatura.

La siguiente fase del modelo es la puesta en común (PC/D) de un ejercicio (E) que se ha proporcionado previamente (concretamente al finalizar la sesión anterior). Este ejercicio, sencillo y de corto alcance, tiene como objetivo que el alumno se enfrente directamente con el problema sin ninguna ayuda y, al menos, identifique las lagunas que existen en sus modelos mentales. Como queremos que este ejercicio estimule el aprendizaje



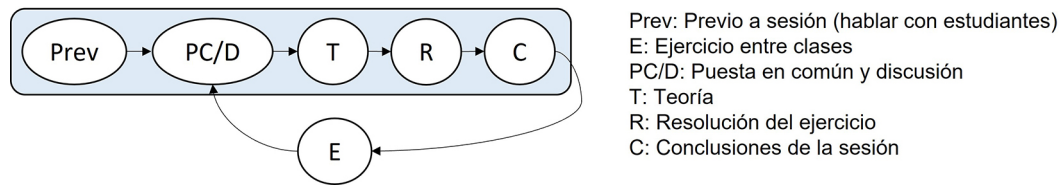


Figura 3. Modelo metodológico posible para el CIMA.

autónomo deberemos dar algunas claves para que, quien quiera, pueda consultar la información necesaria para resolverlo. Sin embargo, se trata más bien de que analicen la necesidad (el hueco en su modelo mental) que vamos a tratar en la siguiente sesión. Aquellos alumnos que hayan trabajado el ejercicio o indagado en el problema podrán exponer sus conclusiones libremente y se establecerá una puesta en común e incluso discusión en el caso de que haya varias soluciones posibles. También se animará a que los alumnos que lo hayan intentado y no hayan llegado a la solución expongan sus problemas o los errores obtenidos, de los cuales aprenderemos todos.

Una vez terminada la puesta en común, se explicarán aquellos contenidos conceptuales que dan respuesta al ejercicio que acabamos de tratar (T). Aunque se vuelve a explicar teoría como en el modelo metodológico actual, ahora está precedida del problema que se resuelve con esa teoría de manera que los alumnos han podido enfrentarse a la necesidad antes de ver las herramientas que la solucionan. Además, creemos que de esta forma puede el alumno reestructurar sus modelos mentales más fácilmente.

Dado que la asignatura es de carácter eminentemente práctico, todas las explicaciones conceptuales se acompañarán de ejercicios que apliquen esos contenidos. Algunos ejercicios se resolverán en común entre todos y otros por parejas. Por último, se resolverá el ejercicio que habían trabajado antes de la sesión aplicando la nueva teoría (R) y se detallarán de manera participativa entre todos los alumnos las conclusiones a las que hemos llegado (C). Al final de la sesión se propondrá el ejercicio que tendrán que trabajar para la próxima sesión.

Secuencia de actividades programadas

El CIMA cubre 8 sesiones de 1 hora y 50 minutos de duración lo que hace un total aproximado de 15 horas. A continuación, se presenta la secuencia de actividades de cada una de las sesiones (tabla 1). Todas ellas tienen la misma estructura fundamentada en el modelo metodológico posible de la figura 3 pero cambiando los contenidos conceptuales y los ejercicios resueltos.



Tabla 1. Secuencia de actividades de las sesiones que componen el CIMA

Actividad	Descripción	Tiempo
Sesión 1 – Codificación de la información		
Previo	Conversación informal para construcción del entorno de aprendizaje seguro	10 minutos
Cuestionario inicial	Al inicio del CIMA se distribuye el cuestionario que nos permitirá evaluar las evoluciones en los modelos mentales de los estudiantes (ver apartado siguiente)	20 minutos
Teoría y ejercicios	Explicación de los contenidos conceptuales esenciales sobre la codificación de la información y ejercicios ilustrativos de aplicación de los contenidos	60 minutos
Conclusiones	Establecimiento de las conclusiones a la sesión (participativo)	15 minutos
Próxima sesión	Distribución del ejercicio para próxima clase	5 minutos
Sesión 2 – Tipos de datos		
Previo	Conversación informal para construcción del entorno de aprendizaje seguro	10 minutos
Puesta en común de ejercicio	Resolución participativa del ejercicio trabajado entre clases y aclaración de problemas	10 minutos
Discusión y conclusiones	Conclusiones sobre el ejercicio orientadas para enlazar con la teoría que se trata a continuación	10 minutos
Teoría y ejercicios	Explicación de los contenidos conceptuales esenciales sobre los tipos de datos en C y ejercicios ilustrativos de aplicación de los contenidos	60 minutos
Conclusiones	Establecimiento de las conclusiones a la sesión (participativo)	15 minutos
Próxima sesión	Distribución del ejercicio para próxima clase	5 minutos
Sesión 3 – Operadores básicos		
Previo	Conversación informal para construcción del entorno de aprendizaje seguro	10 minutos
Puesta en común de ejercicio	Resolución participativa del ejercicio trabajado entre clases y aclaración de problemas	10 minutos
Discusión y conclusiones	Conclusiones sobre el ejercicio orientadas para enlazar con la teoría que se trata a continuación	10 minutos
Teoría y ejercicios	Explicación de los contenidos conceptuales esenciales sobre los operadores básicos en C y ejercicios ilustrativos de aplicación de los contenidos	60 minutos
Conclusiones	Establecimiento de las conclusiones a la sesión (participativo)	15 minutos
Próxima sesión	Distribución del ejercicio para próxima clase	5 minutos
Sesión 4 – Operadores avanzados		
Previo	Conversación informal para construcción del entorno de aprendizaje seguro	10 minutos



Puesta en común de ejercicio	Resolución participativa del ejercicio trabajado entre clases y aclaración de problemas	10 minutos
Discusión y conclusiones	Conclusiones sobre el ejercicio orientadas para enlazar con la teoría que se trata a continuación	10 minutos
Teoría y ejercicios	Explicación de los contenidos conceptuales esenciales sobre los operadores avanzados en C y ejercicios ilustrativos de aplicación de los contenidos	60 minutos
Conclusiones	Establecimiento de las conclusiones a la sesión (participativo)	15 minutos
Próxima sesión	Distribución del ejercicio para próxima clase	5 minutos
Sesión 5 – Estructuras de control		
Previo	Conversación informal para construcción del entorno de aprendizaje seguro	10 minutos
Puesta en común de ejercicio	Resolución participativa del ejercicio trabajado entre clases y aclaración de problemas	10 minutos
Discusión y conclusiones	Conclusiones sobre el ejercicio orientadas para enlazar con la teoría que se trata a continuación	10 minutos
Teoría y ejercicios	Explicación de los contenidos conceptuales esenciales sobre las estructuras de control en C y ejercicios ilustrativos de aplicación de los contenidos	60 minutos
Conclusiones	Establecimiento de las conclusiones a la sesión (participativo)	15 minutos
Próxima sesión	Distribución del ejercicio para próxima clase	5 minutos
Sesión 6 – Funciones		
Previo	Conversación informal para construcción del entorno de aprendizaje seguro	10 minutos
Puesta en común de ejercicio	Resolución participativa del ejercicio trabajado entre clases y aclaración de problemas	10 minutos
Discusión y conclusiones	Conclusiones sobre el ejercicio orientadas para enlazar con la teoría que se trata a continuación	10 minutos
Teoría y ejercicios	Explicación de los contenidos conceptuales esenciales sobre las funciones en C y ejercicios ilustrativos de aplicación de los contenidos	60 minutos
Conclusiones	Establecimiento de las conclusiones a la sesión (participativo)	15 minutos
Próxima sesión	Distribución del ejercicio para próxima clase	5 minutos
Sesión 7 – Punteros I		
Previo	Conversación informal para construcción del entorno de aprendizaje seguro	10 minutos
Puesta en común de ejercicio	Resolución participativa del ejercicio trabajado entre clases y aclaración de problemas	10 minutos
Discusión y conclusiones	Conclusiones sobre el ejercicio orientadas para enlazar con la teoría que se trata a continuación	10 minutos



Teoría y ejercicios	Explicación de los contenidos conceptuales esenciales sobre punteros en C y ejercicios ilustrativos de aplicación de los contenidos	60 minutos
Conclusiones	Establecimiento de las conclusiones a la sesión (participativo)	15 minutos
Próxima sesión	Distribución del ejercicio para próxima clase	5 minutos
Sesión 8 – Punteros II		
Previo	Conversación informal para construcción del entorno de aprendizaje seguro	10 minutos
Puesta en común de ejercicio	Resolución participativa del ejercicio trabajado entre clases y aclaración de problemas	10 minutos
Discusión y conclusiones	Conclusiones sobre el ejercicio orientadas para enlazar con la teoría que se trata a continuación	10 minutos
Teoría y ejercicios	Explicación de los contenidos conceptuales avanzados sobre punteros en C y ejercicios ilustrativos de aplicación de los contenidos	60 minutos
Conclusiones	Establecimiento de las conclusiones a la sesión (participativo)	5 minutos
Cuestionario final	Al finalizar el CIMA se distribuye el cuestionario que nos permitirá evaluar las evoluciones en los modelos mentales de los estudiantes (ver apartado siguiente)	15 minutos

Aplicación del CIMA

Relato resumido de las sesiones

El desarrollo de las sesiones se ha realizado sin inconvenientes y la temporización de las actividades ha resultado adecuada para el correcto desempeño de estas. En general el desempeño de las actividades ha sido satisfactorio y la actuación docente se ha mantenido fiel a la secuencia planeada y los principios metodológicos que motivan el CIMA. El hecho de haber identificado en el mapa los contenidos conceptuales esenciales ha permitido centrarnos en ellos, dedicándoles más tiempo y dejando de lado otros menos importantes para que los estudien los alumnos por su cuenta.

Aunque hemos recibido un *feedback* satisfactorio por parte de los alumnos para la mayoría de las actividades, otras no han sido tan bien acogidas quizás porque los contenidos eran más arduos o difíciles de asimilar con los esquemas mentales actuales de los estudiantes.

El clima de trabajo se puede caracterizar de positivo y la interacción de los alumnos ha sido adecuada con una participación alta en líneas generales. No obstante, pasadas varias sesiones algunos alumnos reconocen que no se están enterando de mucho pero solo lo refieren cuando se



les pregunta directamente. Este hecho confirma que no conseguimos llegar a todos los alumnos con la prerrogativa del entorno de aprendizaje seguro y la confianza de que pedirán ayuda de forma proactiva en caso de que lo necesiten.

Evaluación del aprendizaje de los estudiantes

Como hemos comentado, esta asignatura es del primer curso del Grado en Ingeniería de las Tecnologías de Telecomunicación. Por ello, la experiencia nos dice que la mayoría del alumnado que inicia el curso no tienen ningún modelo mental acerca de la programación, y muchos menos, de las herramientas y el lenguaje que vamos a utilizar.

En la tabla 2 se muestra la herramienta que utilizaremos para evaluar la evolución de los modelos mentales de los estudiantes. Se trata de un cuestionario compuesto por diez preguntas de muy diversa índole. El cuestionario se distribuyó en la primera sesión del CIMA permitiendo modificar los contenidos y la secuencia de actividades en función del punto de partida de los alumnos. Como el CIMA ha sido diseñado para un alumnado con escasos o nulos conocimientos en la materia, los resultados del cuestionario inicial podrían haber afectado a la metodología si hubiésemos identificado alumnos con conocimientos previos en la materia. En ese caso, no se hubiera modificado la secuencia de actividades, sino que se hubieran propuesto entre sesiones ejercicios de diversa dificultad para que todos los alumnos pudieran avanzar en el aprendizaje, independientemente de si tenían conocimientos previos o no. No fue este el caso. Los pocos alumnos que demostraron al inicio del CIMA que poseían conocimientos previos fueron alumnos repetidores y los conocimientos no eran siempre completamente acertados.

Entre las preguntas del cuestionario incluimos una (la primera) que cuestiona la utilidad que el alumno cree que tiene la programación en su futuro con el objetivo de comprobar si esta percepción se modifica a medida que avanzamos en el aprendizaje. Por otro lado, las preguntas 2, 5, 6 y 7 están relacionadas con los contenidos conceptuales de la asignatura, mientras que las preguntas 3, 4, 8, 9 y 10 presentan aspectos prácticos que ponen a prueba los contenidos actitudinales y procedimentales de los alumnos.

Como se ha indicado en la secuencia de actividades, en la última sesión del CIMA se distribuyó de nuevo el mismo cuestionario. Una vez finalizado el CIMA se analizó la evolución de los modelos mentales de los estudiantes comparando los resultados de los cuestionarios inicial y final que se plasman en la escalera de aprendizaje de la figura 4. En particular obtuvimos 32 alumnos que respondieron tanto al cuestionario inicial



Tabla 2. Cuestionario para el seguimiento de la evolución de los modelos mentales de los estudiantes

1	De 1 a 5, ¿cómo de útil crees que es la programación en tu futuro? (1 – Muy importante / 5 – Nada importante)
2	Nombra dos características del lenguaje C
3	Diseña el algoritmo de un semáforo que esté 20 segundos en verde y el resto del tiempo en rojo
4	Para resolver algunos problemas solo puede utilizarse un lenguaje de programación específico, no vale cualquiera. ¿Verdadero o falso?
5	¿Cuántos tipos de datos hay en C?
6	¿Cuál es la diferencia entre for y while?
7	¿Cuál es la diferencia entre while y do while?
8	¿Cuántas variables y constantes hay en el siguiente ejemplo? <pre>#define PARAM 4 int main() { int numero = 0; char carac = '@'; if (numero < PARAM) printf («%c», carac+numero); return 0; }</pre>
9	¿Qué imprime el ejemplo anterior?
10	¿Qué devuelve la expresión if (numero < PARAM)?

como al final identificándose con el mismo nombre en ambos casos. En la figura 4 se resumen los resultados identificando cuatro estadios en el aprendizaje: desconocimiento, respuesta errónea, respuesta correcta e incompleta, y respuesta correcta y completa.

Cada fila agrupa los resultados de una pregunta (de la 2 a la 10). Cada figura representa 5 alumnos correspondiendo las rojas al cuestionario pre y las verdes al cuestionario post. De la figura podemos extraer que una amplia mayoría de los alumnos poseían un desconocimiento casi completo de los contenidos conceptuales y procedimentales de la asignatura antes de la aplicación del CIMA. Tras la experimentación, amplios grupos de estudiantes han avanzado hasta el último tramo de la escalera. No obstante, cabe destacar que tras la aplicación del CIMA, algunos alumnos siguen en la etapa del desconocimiento, lo que debe ser motivo de análisis para futuras actuaciones.

Por otro lado, en la figura 5 se muestran los resultados de la primera pregunta del cuestionario que pretendía cuantificar el interés de los alumnos en la asignatura antes del CIMA y tras la aplicación de este.



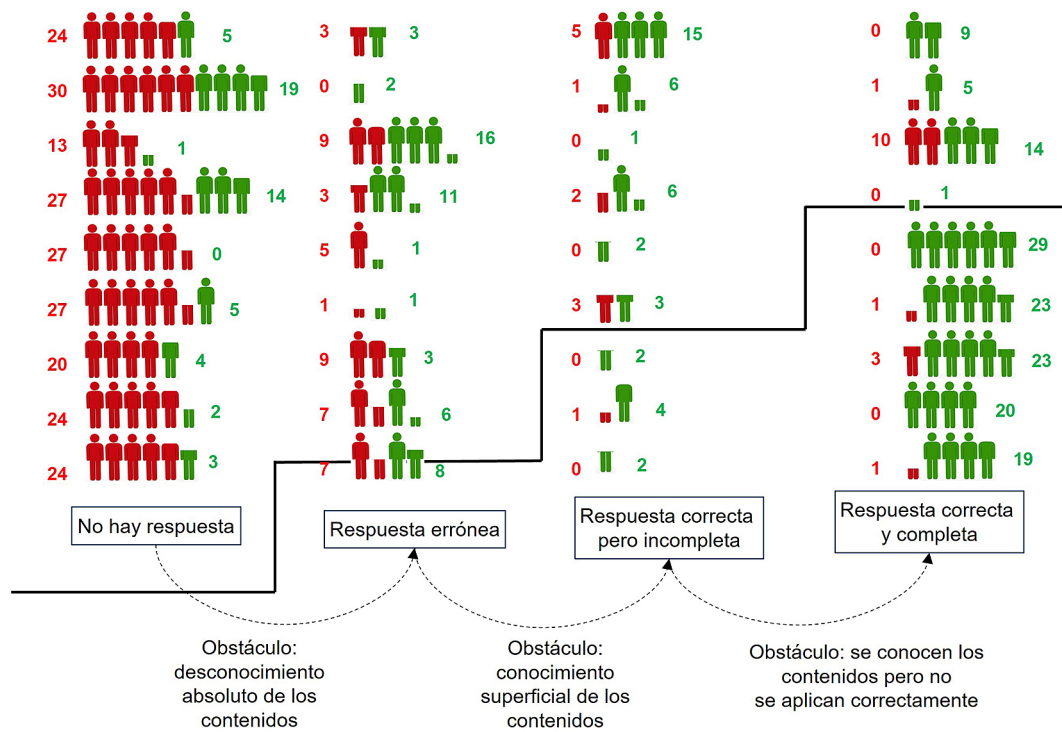


Figura 4. Resultados de aprendizaje a partir del cuestionario pre-/post-CIMA.

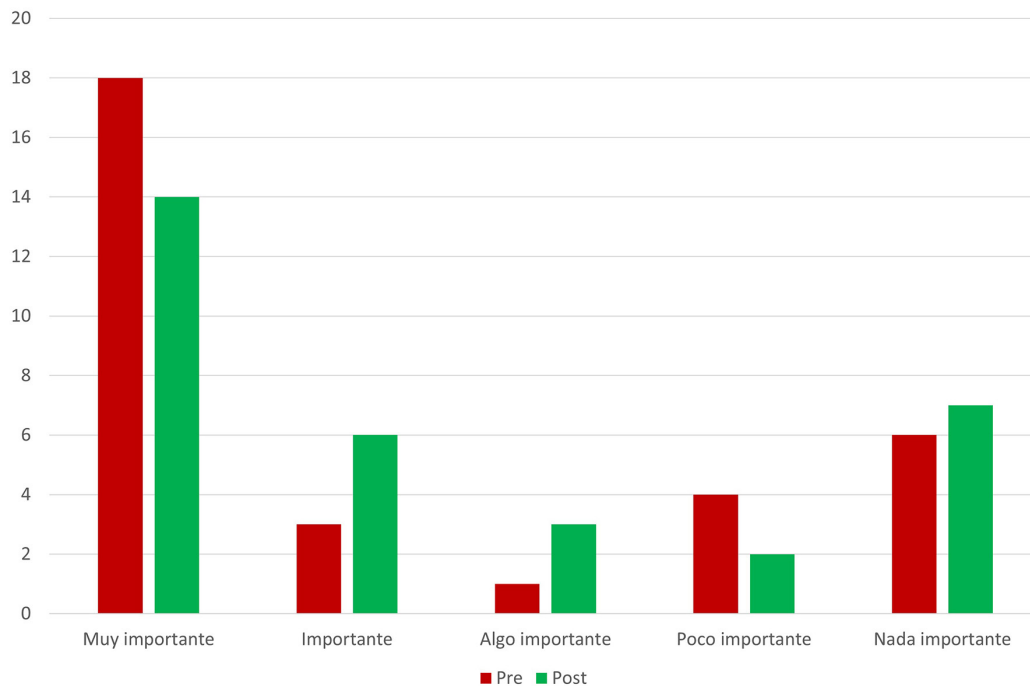


Figura 5. Interés de los alumnos a partir del cuestionario pre-/post-CIMA.



Evaluación del CIMA

Cuestiones a mantener y cambios a introducir para un futuro Ciclo de Mejora

Entre los aspectos destacados cuyo éxito anima a mantener en el futuro están el mapa de contenidos (que fue realizado para toda la asignatura aunque el CIMA solo cubría la mitad de ella), el uso del mapa para que los estudiantes entiendan cómo se relacionan entre sí los contenidos, la actividad previa a las sesiones en las que construimos el entorno seguro de aprendizaje a través de una interacción distendida con los estudiantes, los ejercicios entre sesiones que permiten a los estudiantes avanzar de forma autónoma o plantearse los problemas para la siguiente sesión. Los ejercicios entre clases deberían enviarse a los estudiantes de manera más formal en vez de únicamente proyectarlos en clase para que puedan acceder a ellos, aunque no hayan participado de la última sesión.

Con respecto a los cambios que deberíamos introducir en el futuro cabe destacar el refuerzo de aquellos contenidos cuyas actividades relacionadas han sido menos satisfactorias, así como el hecho de cubrir más contenidos de la asignatura con un Ciclo de Mejora más amplio.

Aspectos de la experiencia que se pretenden incorporar a toda la práctica docente habitual

Si algo se puede concluir de esta experimentación es que *el aprendizaje no tiene por qué estar exclusivamente en manos del docente y desarrollado dentro del aula*. Los estudiantes deben aprender a ser proactivos en su aprendizaje y, para ello, nada mejor que los ejercicios que se han propuesto entre sesiones. Como se ha comentado, no son ejercicios que aplican los contenidos conceptuales ya tratados, sino que proponen a los estudiantes pequeños retos que solo pueden ser resueltos analizando el problema, descubriendo las lagunas que aún poseen en los contenidos conceptuales e intentando suplirlas de manera autónoma recurriendo a los materiales de la asignatura u otras fuentes. Con este método se consiguen tres objetivos. Primero, *los estudiantes aprenden dentro de su zona de desarrollo próximo* (Vygotsky, 1987) sin necesidad de contar con la tutela del docente como fuente de contenidos. Segundo, *los estudiantes desarrollan contenidos procedimentales y actitudinales claves de la asignatura (pensamiento crítico, razonamiento, etc.)*. Finalmente, si los contenidos conceptuales más asequibles pueden ser aprendidos autónomamente por los estudiantes, *las sesiones pueden dedicarse con mayor profundidad a los contenidos más complejos*.



Principios Didácticos argumentados que han guiado la experiencia presente y que deben permanecer en el futuro

El principio didáctico que debe permanecer en el futuro es el de *espacio de aprendizaje seguro*, pero haciéndolo más patente, quizás verbalizándolo más. Creemos que construir en el aula un ambiente relajado de aprendizaje donde el error sea consustancial al proceso y los estudiantes puedan hablar libremente, expresar sus ideas y opiniones, sin miedo a ser recriminados, señalados o calificados por el docente o sus compañeros, es uno de los grandes principios didácticos que pueden suponer un cambio sustancial en el aprendizaje.

Durante esta experimentación se ha intentado crear ese espacio de aprendizaje seguro, aunque el resultado ha distado del ideal que nos habíamos propuesto. La mayoría de los estudiantes han respondido satisfactoriamente frente al docente, pero les avergüenza hacerlo entre sus iguales, supongo que por causas sociales. Sin eliminar esa barrera en las mentes de los estudiantes es complicado que en público se atrevan a equivocarse o a admitir que desconocen algo. Por eso este principio debe permanecer en el futuro, pero merece una reflexión más profunda el cómo fortalecerlo para ponerlo en práctica con mayor éxito.

Referencias bibliográficas

- Bain, K. (2007). *Lo que hacen los mejores profesores universitarios*. Universitat de València. Servei de Publicacions.
- Calvillo-Arbizu, J. (2018). Animar al aprendizaje autónomo a través de gamificación en programación orientada a objetos. *Jornadas de Formación e Innovación Docente del Profesorado 1*, 80-95. <http://dx.doi.org/10.12795/JDU.2018.i01.04>.
- Delord, G.; Hamed, S.; Porlán, R. y De Alba, N. (2020). Los Ciclos de Mejora en el Aula. En N. De Alba y R. Porlán (Coord.), *Docentes universitarios. Una formación centrada en la práctica* (pp. 128-162). Ediciones Morata.
- Finkel, D. (2008). *Dar clase con la boca cerrada*. Universitat de València. Servei de Publicacions.
- Porlán, R. (Coord.) (2017). *Enseñanza universitaria. Cómo mejorarla*. Ediciones Morata.
- Vygotsky, L. S. (1987). *El desarrollo de los procesos psicológicos superiores*. Austral.



