

# Enseñanza de los algoritmos de programación exhaustivos desde un modelo docente constructivista

## Teaching of the exhaustive programming algorithms from a constructivist educational model

Andrés Jiménez Ramírez

ORCID: <https://orcid.org/0000-0001-8657-992X>

Universidad de Sevilla

Departamento de Lenguajes y Sistemas Informáticos

[ajramirez@us.es](mailto:ajramirez@us.es)

DOI: <http://dx.doi.org/10.12795/9788447222865.139>

Pp.: 2439-2456



## Resumen

Dentro del Grado de ingeniería informática – ingeniería de computadores se ha desarrollado un ciclo de mejora de 8 horas donde se enseña la programación con algoritmos exhaustivos dentro de la asignatura Análisis y Diseño de Datos y Algoritmos. Este ciclo de mejora contempla un proceso metodológico donde las actividades de ideación y debate toman especial relevancia. El artículo contiene el mapa de contenidos y los cuestionarios inicial y final que sirven para comprobar si se ha producido aprendizaje en los alumnos. Tanto las sensaciones del profesor como los resultados de las escaleras de aprendizaje fueron positivos, aunque se identificaron punto de mejora para futuros ciclos de mejora.

*Palabras clave:* Análisis y diseño de datos y algoritmos, Ingeniería del Software, Experimentación docente universitaria, Innovación docente, Algoritmos exhaustivos.

## Abstract

Within the Degree of Software Engineering – Computer Engineering, an improvement cycle of 8 hours has been developed where programming with exhaustive algorithms is taught within the subject Analysis and Design of Data and Algorithms. This improvement cycle contemplates a methodological process where ideation and discussion activities take special relevance. The article contains the content map and the initial and final questionnaires that are used to check if the students' learning has taken place. Both the teacher's feelings and the results of the learning ladders were positive, although points for improvement were identified for future improvement cycles.

*Keywords:* Analysis and design of data and algorithms, Software engineering, University teaching experimentation, Teaching innovation, Exhaustive algorithms.



## Descripción del contexto

Esta asignatura, Análisis y Diseño de Datos y Algoritmos, se imparte en 2º del Grado de Ingeniería de Computadores. Las sesiones son los lunes a las 8:30 y son de 1 hora y 50 minutos de duración. El grupo se compone de unos 15 alumnos. La mayoría de los alumnos están presentes desde el principio de la clase, pero algunos van llegando durante la primera media hora.

Este ciclo comienza con la asignatura ya avanzada e incluye 4 sesiones. El tema de «Algoritmos genéticos» se terminó en la clase anterior dejando planteado un ejercicio para la siguiente clase. En la primera sesión se plantea enseñar a resolver ese problema concluyendo el tema de «Algoritmo Genéticos» y empezar con el tema de «Programación Dinámica y Backtracking» que se mantendrá durante las siguientes 4 sesiones.

El resto del documento se organiza como sigue. En la sección 1 se presenta el diseño del ciclo de mejora para lo cual se han tomado como referencia algunos estudios en el campo de la innovación en la docencia universitaria (Rivero y Porlán, 2017; Finkel 2008) y experiencias similares realizada por profesores de la Universidad de Sevilla (Navarro Medina y Porlán, 2020). En la sección 2 se describe la aplicación del propio ciclo de mejora. Finalmente, la sección 3 incluye la evaluación personal que ha surgido de realizar esta experiencia.

## Diseño previo del CIMA

### Mapa de contenidos

El mapa de contenidos de la figura 1 se describe con este resumen:

- Para resolver problemas del tipo del caballo o la mochila hay que atacar dos objetivos: (1) Generalizar el problema y (2) modelar el algoritmo.
- Generalizar implica decidir qué atributos tiene el problema e indicar cuáles los definen y cuáles están relacionados con el tamaño de este.
- Para modelar el problema, hemos de saber si buscamos la mejor solución, o una aproximación. Aquí hay que pensar sobre el tiempo, la memoria y la precisión de los diferentes algoritmos.
- Para los algoritmos exhaustivos tenemos tres técnicas similares en las cuales hay que discutir si podemos usar heurísticas para que el algoritmo vaya más rápido. En el caso concreto en que la generalización resultase compleja, solo se podría usar la técnica de programación dinámica.
- Para los algoritmos aproximados, vemos la técnica voraz que es una simplificación de Backtracking.



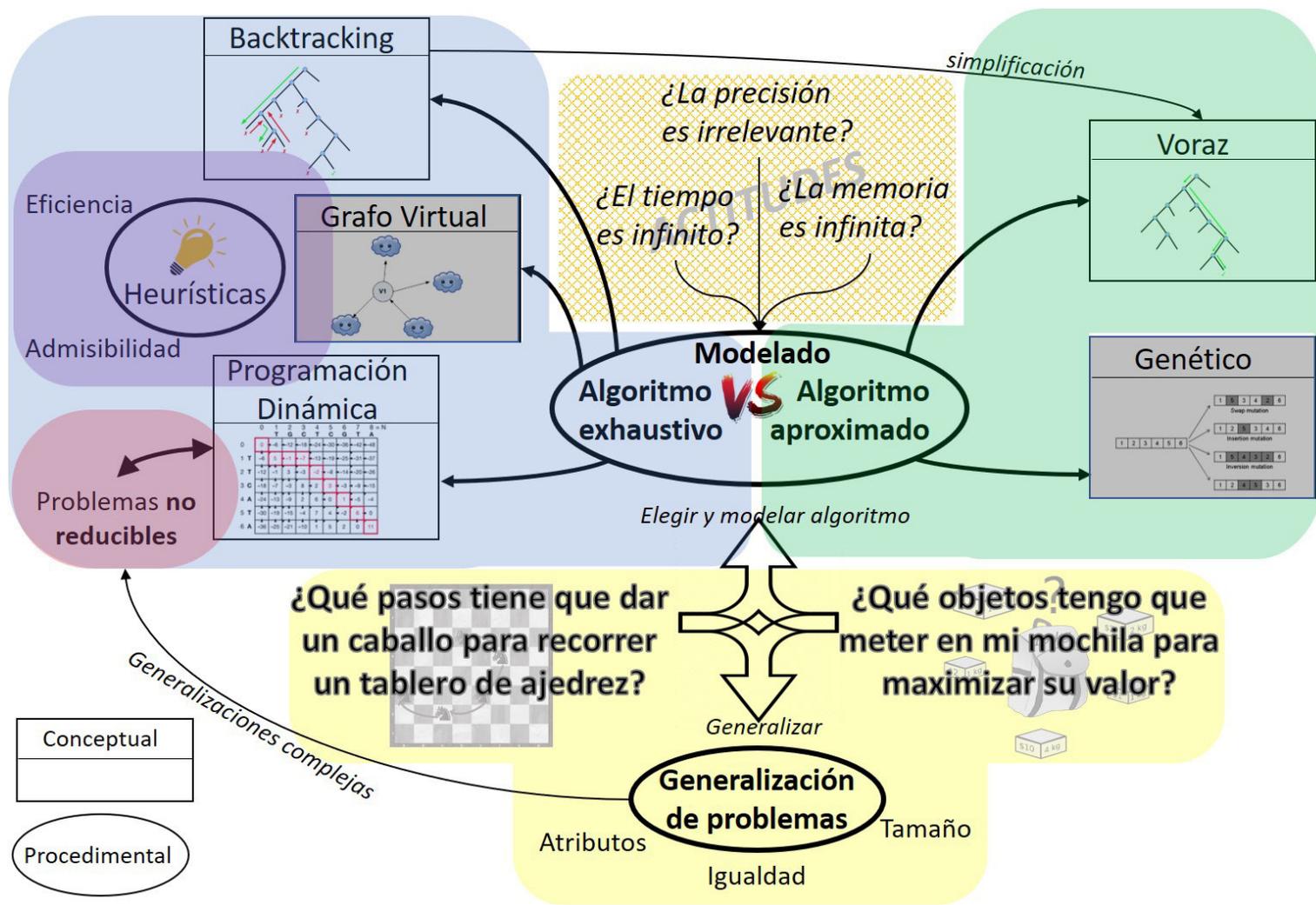


Figura 1. Mapa de contenidos.



Más precisamente, el anterior mapa muestra 4 grandes bloques de color. En primer lugar, en el bloque amarillo, se muestran las 2 preguntas/problemas principales que rigen el mapa y que motivan los dos contenidos procedimentales principales (los óvalos): la generalización de problemas y el modelado de algoritmos. Por un lado, con respecto a la generalización de problemas, se plantean tres conceptos factuales que, a su vez, están relacionados con este contenido procedimental: hay que saber cuáles son los «atributos» en la generalización y cuáles tiene que usarse para definir el concepto de «igualdad» y «tamaño» de problemas. Por otro lado, con respecto al modelo de algoritmos, se plantea el desafío de saber decidir cuándo se ha de usar uno exhaustivo frente a uno aproximado.

En segundo lugar, en el bloque azul, se muestran los contenidos relacionados concretamente con los algoritmos exhaustivos. Los tres algoritmos que se estudian en la asignatura, y que se entienden como contenido conceptual (las cajas), son los grafos virtuales, el Backtracking (también llamado Vuelta atrás) y la programación dinámica (Kennedy, 1986; Golomb y Baumert, 1965). Hay un concepto avanzado que solo se aplica a la programación dinámica que son los «problemas no reducibles», de ahí que su sombra roja solo solape con este tipo de algoritmo. Con respecto a los tres algoritmos anteriores, se plantea como contenido procedimental, el saber cómo generar heurísticas y, dentro de este contenido, son relevantes los conceptos de «admisibilidad» y «eficiencia» de la heurística. El hecho de que el algoritmo de grafos virtuales este sombreado en gris indica que ese contenido es parte de otra parte de la asignatura, aunque está relacionada con esta.

En tercer lugar, en el bloque verde se visualizan los algoritmos aproximados, tan solo el algoritmo voraz se ve en esta parte de la asignatura, el algoritmo genético se ve en otro momento y por eso está sombreado.

Por último, en el bloque naranja rayado con el texto «Actitudes» de fondo se han dejado aquellos conceptos actitudinales que son relevantes para esta parte de la asignatura y que se aplican, sobre todo, al hecho de elegir un tipo de algoritmo a la hora de resolver un problema dado.

### **Modelo metodológico**

El modelo metodológico de la figura 2 tiene como base el modelo metodológico constructivista (Bain, 2007), y potencia más el trabajo de ideación del alumno previo a la muestra de la solución posible por parte del docente, de manera que permita generar actividades de contraste.



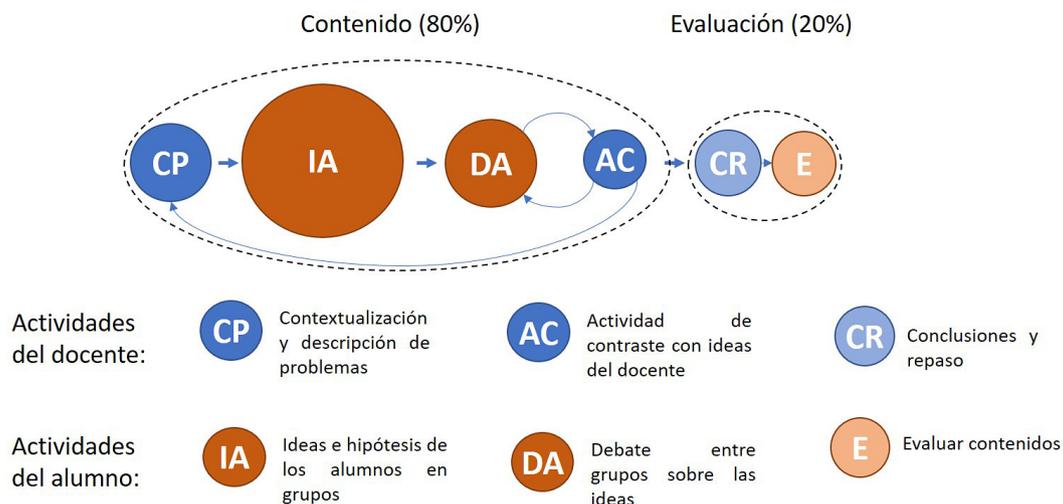


Figura 2. Modelo metodológico posible.

### Secuencia de actividades

La secuencia detallada de las actividades que se proponen para este ciclo de mejora se divide en 4 días como se muestran en las tablas 1-4. Es importante destacar que, aunque las clases fueran de 1 hora y 50 minutos, no todos los días suman dicha cantidad por tener algún margen de tiempo.

Tabla 1. Secuencia de actividades del día 1

1		Cuestionario Inicial	20 min
Los alumnos responden individualmente el cuestionario, preguntando sus dudas si hace falta.			
Conocer el punto de partida del alumno. Que el alumno reflexione.			
2	CP	Contextualización: programación dinámica y problema de la mochila	25 min
Presentar el tema actual y contar los orígenes que llevan a la programación dinámica. Así como describir cómo se puede plantear la resolución del problema de la mochila desde esa perspectiva.			
Que los alumnos vean cómo se idea el algoritmo de programación dinámica y prepararlos para la abstracción metodológica.			
Programación Dinámica, Problema de la mochila y Generalización de problemas.			
3	IA	Ideación del alumno a través de taller conceptual	45 min
Realizar el taller: «creación de mi algoritmo de programación dinámica» en grupos.			
Que los alumnos hagan suyo este algoritmo y recapaciten sobre sus detalles.			
Programación Dinámica y Generalización de problemas			



4	DA-AC	Debate entre alumnos con incorporación de ideas del profesor	20 min
Exposición por grupos de las conclusiones del taller y debate entre ellos. El profesor va reconduciendo la discusión hacia los puntos más conflictivos y poniendo en sobre el debate, si no sale, contenidos actitudinales.			
Que los alumnos se esfuercen en defender sus algoritmos y aprendan las debilidades detectadas por otros compañeros además de recapacitar sobre sus ideas preconcebidas.			
Programación Dinámica, Generalización de problemas y Actitudes sobre precisión, tiempo y memoria.			

Tabla 2. Secuencia de actividades del día 2

5	CR	Conclusiones de la programación dinámica	10 min
Conclusión, por parte del profesor, de los elementos más relevantes discutidos anteriormente.			
Que los alumnos verifiquen y validen lo aprendido en el taller.			
Programación Dinámica, Generalización de problemas y Actitudes sobre precisión, tiempo y memoria.			
6	E	Evaluación	15 min
Evaluación rápida sobre conceptos clave.			
Que los alumnos verifiquen y validen lo aprendido en el taller.			
7	CP	Contextualización: Marco de trabajo en la asignatura	45 min
Describir cómo se trabaja en la asignatura con los algoritmos exhaustivos como la programación dinámica. Introducir también los componentes de ese marco de trabajo para Backtracking.			
Que los alumnos asimilen una forma de trabajar común que es la que se usa en la asignatura y, además, entiendan los conceptos básicos de Backtracking y el problema del caballo.			
Programación Dinámica, Backtracking y Modelado.			
8	IA	Ideación de los alumnos sobre modelado	35 min
Organizar a los alumnos en grupos para que trabajen sobre el modelado de los algoritmos para el problema del caballo y de la mochila usando el marco de trabajo de la asignatura.			
Que los alumnos entiendan que ambos problemas son similares cuando se abstraen con modelos muy similares.			
Programación Dinámica, Backtracking y Modelado.			



Tabla 3. Secuencia de actividades del día 3

9	DA-AC	Debate entre alumnos con incorporación de ideas del profesor	30 min
Discusión sobre las diferentes alternativas de modelado de problemas. El profesor va reconduciendo la discusión hacia los puntos más conflictivos y poniendo sobre el debate, si no sale, contenidos actitudinales.			
Que los alumnos vean que hay diferentes alternativas a la hora de modelar y que entiendan que afectan a la velocidad y consumo de memoria.			
Modelado, Actitudes sobre tiempo y memoria.			
10	CP	Contextualización sobre otros posibles problemas	20 min
Descripción de otros problemas similares al del caballo y la mochila que pueden ser resueltos mediante Backtracking y programación dinámica (e.g., el de los saltos o las jarras, ver Anexo). Hacer ver, además, que estos algoritmos repiten demasiados cálculos.			
Poner a los alumnos en predisposición para ver esos problemas como ejemplos particulares de los anteriores.			
Modelado, Generalización de problemas, Actitudes sobre tiempo y memoria.			
11	IA	Ideación sobre modelado y eficiencia	30 min
Organizar a los alumnos en grupo para que resuelvan el problema de las jarras y los saltos y, además, para que busquen dónde los algoritmos de Backtracking y programación dinámica repiten más cálculos, proponiendo alternativas para evitarlo.			
Que los alumnos se ejerciten en diferentes generalizaciones de problemas y modelado de algoritmos y piensen por primera vez en heurísticas que puedan aumentar la eficiencia.			
Modelado, Generalización de problemas, Heurísticas, Actitudes sobre tiempo y memoria.			
12	DA-AC	Debate entre alumnos con incorporación de ideas del profesor	20 min
Discusión sobre las generalizaciones y modelos y, luego, sobre las propuestas de heurísticas. El profesor va reconduciendo la discusión hacia los puntos más conflictivos y poniendo sobre el debate, si no sale, contenidos actitudinales.			
Que los alumnos vean que hay diferentes alternativas a la hora de modelar y también a la hora de proponer heurísticas.			
Modelado, Generalización de problemas, Heurísticas, Actitudes sobre tiempo y memoria.			

Tabla 4. Secuencia de actividades del día 4

13	CP	Contextualización: Algoritmos aproximados	15 min
Describir contextos donde la precisión no es tan relevante como la velocidad, planteado como un problema en sí.			
Que los alumnos entiendan que los algoritmos exhaustivos no son la «panacea» y deben existir otros.			
Algoritmos aproximados, Actitudes sobre precisión y tiempo.			
14	IA	Ideación de los alumnos sobre algoritmos voraces	35 min
Organizar a los alumnos en grupos para que trabajen sobre posibles modificaciones del algoritmo de Backtracking para que consigan obtener soluciones aproximadas para los problemas anteriores sin tener que cambiar la generalización de estos.			



Que los alumnos hagan suyo el algoritmo voraz y lo vean como una simplificación del Backtracking.			
Voraz.			
15	DA-AC	Debate entre alumnos con incorporación de ideas del profesor	20 min
Discusión sobre las diferentes alternativas de algoritmos voraces. El profesor va reconduciendo la discusión hacia los puntos más conflictivos y poniendo sobre el debate, si no sale, contenidos actitudinales.			
Que los alumnos puedan ver y evaluar diferentes alternativas de algoritmos voraces junto a sus limitaciones.			
Voraz, Actitudes sobre precisión, tiempo y memoria.			
16	CR	Conclusiones de algoritmos exhaustivos y aproximados	20 min
Conclusión, por parte del profesor, de los elementos más relevantes de los algoritmos exhaustivos de las clases anteriores y los aproximados de la clase de hoy.			
Que los alumnos verifiquen y validen lo aprendido.			
Todo.			
19	E	Cuestionario final	20 min
Los alumnos responden individualmente el cuestionario sin hacer preguntas.			
Conocer el punto de nivel de aprendizaje al que han llegado los alumnos.			

### Cuestionario Inicial y Final

Con el objetivo de valorar si se ha producido aprendizaje por parte del alumno, se pretende realizar un cuestionario al principio y al final de las actividades planteadas. Dichas preguntas están formuladas de tal manera que el alumno se siente cómodo a la hora de responder. Para ello, se formulan en lenguaje coloquial y, además, pueden usar un pseudónimo a la hora de responder. Dicho pseudónimo sirve para anonimizar las respuestas del alumno, pero permite un análisis individual *a posteriori*.

Las preguntas que se plantean para los cuestionarios inicial y final son las siguientes:

1. *¿Cómo sería un algoritmo que resolviera un problema como «los pasos que tiene que dar un caballo para que visite todas las casillas de un tablero sin repetir las»?*
2. *¿Cuántos pasos daría tú algoritmo para un tablero de 4x3? Argumentalo*
3. *¿Qué opciones de moverse tiene el caballo en cada paso?*
4. *Si conoces el nombre de algunos algoritmos que permitan resolver problemas como este, nómbralos o descríbelos si no te acuerdas del nombre.*
5. *¿Se te ocurre cómo podrías adaptar ese algoritmo para el problema de «maximizar el valor de una mochila, considerando un conjunto*



*de objetos, cada uno con su peso, valor y multiplicidad máxima y sabiendo que la mochila tiene cierto peso máximo permitido»?*  
 Arguméntalo.

6. *¿Cuántos pasos tendría como mínimo ahora?, ¿por qué?*
7. *¿Se te ocurre cómo hacer que el algoritmo sea más rápido? Explícalo*
8. *Si me valiera una solución buena, aunque no máxima, ¿se te ocurre como hacerlo aún más rápido?*

## Aplicación del CIMA

### Desarrollo de las sesiones

#### Sesión 1

Comencé esta primera sesión describiendo el nuevo proceso que íbamos a seguir en este bloque de la asignatura, manifestando mi compromiso por realizarlo de manera rigurosa. Del mismo modo, les pedí que se comprometieran a participar activamente en el desarrollo ya que, como recomienda Bain (2007), les iba a pedir que razonaran mucho durante las siguientes cuatro sesiones. Me sorprendió su actitud positiva ante esta innovación. Tras realizar el cuestionario inicial, pasé a describirles la base principal de un algoritmo de programación recursivo –algo ya estudiado en la asignatura– donde se deben plantear alternativas a la hora de crear los problemas recursivos para, finalmente, seleccionar la alternativa que conduzca a la mejor solución, i.e., las bases de la programación dinámica (Kennedy, 1986). El grueso de la clase se invirtió en el taller conceptual (Finkel, 2008) sobre la construcción de un algoritmo propio de programación dinámica donde los alumnos trabajaron por grupos y yo me dediqué a «desatascar» y «relinear» a aquellos grupos que se desviaban del objetivo en sí del taller. Esta actividad, aunque pautaba una serie de tiempos, se dilató más de lo programado. La clase concluyó con un debate moderado por mí sobre las ideas a las que cada grupo había llegado en el taller. Aunque la sensación fue positiva, los alumnos no llegaron a producir ese algoritmo que se buscaba.

#### Sesión 2

Dado que en la primera sesión dio la sensación de falta de tiempo en las conclusiones, esta segunda comenzó con las conclusiones sobre el taller lo cual abrió un pequeño mini debate sobre esas cuestiones principales: la memoria que se consume y el tiempo que emplearía el algoritmo. Tras concluir, realizamos una evaluación en Kahoot sobre los



conceptos del taller. Como aporte teórico de la asignatura, les mostré cómo la programación dinámica y otros algoritmos exhaustivos pueden implementarse dentro del contexto de la asignatura, utilizando los medios que ponemos a su disposición. Resultó gratificante ver cómo los alumnos identificaban cada parte con lo visto en el taller y veían su utilidad. Esta parte llevó más tiempo del esperado ya que los alumnos consultaban mucho más que en años anteriores, debido a que tenían ya mayor conocimiento sobre la temática que ellos mismos había aprendido por sí solos (Vygotsky, 1989).

La clase concluyó volviendo a formar a los alumnos en los mismos grupos que en el taller conceptual, ahora con un objetivo claro: modelar el problema del caballo y la mochila usando las herramientas de la asignatura. Aunque no dio tiempo a que formalizaran correctamente ambos problemas, sí que observé que la mayoría de los grupos demostraba haber entendido los conceptos básicos.

### Sesión 3

En la sesión anterior no dio tiempo a terminar los modelados de ambos problemas por lo que empecé la clase resumiendo los principales inconvenientes o desafíos que yo había observado en la clase anterior, para que continuarán modelando. Pasado un tiempo en que la mayoría de los grupos tenía modelos sólidos les mostré cómo cada grupo había tomado decisiones frente a cómo modelarlo y se abrió un debate sobre como dichos modelados, todos correctos, afectan a la eficiencia del algoritmo. Aunque mi papel era de moderador, aquí si tuve que incorporar ideas porque no llegaban a discutir sobre el hecho de que hay modelados que permiten alcanzar más rápido una mejor solución y, por consiguiente, acabar antes el algoritmo.

La segunda parte de la clase la focalicé sobre la eficiencia. Primero les mostré cómo modelar otros problemas similares como el de las Jarras y los saltos. Junto con la ayuda de los alumnos, se vio como los algoritmos repiten muchos cálculos a la hora de resolver estos y otros problemas. Me sorprendió ver que algunos de los alumnos ya habían identificado previamente esa falta de eficiencia, aunque aún no se habían planteado como resolverla. Esta segunda parte de la clase finalizó con otro tiempo de trabajo en grupo donde tenían que buscar soluciones –las llamadas heurísticas– a ese problema de eficiencia. Finalmente hubo un tiempo de debate dirigido por mi dónde los grupos aportaron sus propuestas. No dio tiempo a concluir por mi parte, pero me resultó positivo ya que los alumnos mantuvieron su inquietud por buscar mejores formas de hacer más eficiente el algoritmo hasta la próxima clase.



## Sesión 4

Empecé la clase trayendo las principales ideas propuestas por los grupos en la sesión anterior y exponiendo situaciones para que razonaran si dichas heurísticas se aplicaban o no. Por ejemplo, se propuso que si el algoritmo no mejoraba una solución durante un tiempo, que se terminara el algoritmo y se diera esa solución por la mejor. Aportando un ejemplo donde la mejor solución está al final, se dieron cuenta de que esa no era una buena heurística o, más concretamente, no era una heurística *admisible*, que era el concepto que se quería enseñar. Este tipo de ideas por parte los alumnos vinieron bien para introducir la siguiente pregunta: ¿Qué es más importante, la precisión o la velocidad? Con esta pregunta los alumnos pudieron entender que hay contextos donde no siempre se busca una mejor solución, sino una solución buena.

En la segunda parte de la sesión los alumnos volvieron a organizarse en grupo para idear una modificación del algoritmo exhaustivo Backtracking (Golomb y Baumert, 1965) que permitiera encontrar una, y solo una, solución que, aunque no fuera la mejor, fuera buena. Me satisfizo ver cómo los alumnos ya tenían ideas sobre cómo querían hacerlo y el debate se produjo rápidamente sin necesidad de imponerlo yo. Mi participación fue para concluir y poder darle un nombre al tipo de algoritmo que estaban ideando, los algoritmos «Voraces».

La sesión concluyó con un resumen general de este bloque de la asignatura y con la realización del cuestionario final, similar al inicial, que me permitió ver cómo los alumnos habían avanzado en su aprendizaje.

### *Evaluación del aprendizaje T14*

El cuestionario fue respondido por 17 personas, aunque algunas de ellas no fueron las mismas entre los dos cuestionarios. Se hicieron un total de 8 preguntas y para cada una de ellas se ha establecido unos niveles de aprendizaje para las respuestas, identificándose, además, los desafíos y obstáculos existentes en cada nivel como se muestra a continuación en las tablas 5-12. En la columna «Nivel» se indica en rojo y en azul los porcentajes de alumnos que estaban en dicho nivel en el momento inicial y final respectivamente.



Tabla 5. Niveles de respuesta y desafíos a pregunta 1

Nivel	Tipo de respuesta	Desafío de aprendizaje para subir de nivel
Nivel 0 12,50 % 0,00 %	No responde o no presenta un algoritmo que pueda resolver el problema del caballo	Entender el problema y la solución buscada
Nivel 1 37,50 % 0,00 %	Plantea un algoritmo que solo permitiría obtener una solución aproximada al problema	Darse cuenta de que es un problema complejo y que hay que hacer una búsqueda
Nivel 2 25,00 % 25,00 %	Aplica una técnica anterior o describe solo una parte del algoritmo	Comprender todas las etapas del algoritmo
Nivel 3 25,00 % 37,50 %	Describe una solución válida, aunque no razona sobre el hecho de tener que probar alternativas en paralelo	Aprender a gestionar problemas con varias alternativas
Nivel 4 0,00 % 37,50 %	Incluye correctamente las alternativas y el hecho de tener que probarlas	

Tabla 6. Niveles de respuesta y desafíos a pregunta 2

Nivel	Tipo de respuesta	Desafío de aprendizaje para subir de nivel
Nivel 0 25,00 % 25,00 %	No responde o no muestra un razonamiento válido	Entender el problema y la solución buscada
Nivel 1 0,00 % 0,00 %	Da una descripción válida, aunque no concreta un valor.	Analizar e imaginar una posible solución al problema
Nivel 2 37,50 % 75,00 %	Da una solución válida, pero no razona sobre el hecho de que puede no haber solución.	Darse cuenta de que no siempre tiene que existir una solución
Nivel 3 37,50 % 0,00 %	Incluye un razonamiento sobre que no haya solución	



Tabla 7. Niveles de respuesta y desafíos a pregunta 3

Nivel	Tipo de respuesta	Desafío de aprendizaje para subir de nivel
Nivel 0 25,00% 25,00%	No responde o no muestra haber entendido los posibles movimientos del caballo.	Entender el problema y la solución buscada
Nivel 1 0,00% 0,00%	Describe correctamente pero no da un valor concreto.	Analizar los detalles de cómo puede avanzar el algoritmo.
Nivel 2 50,00% 50,00%	Solo se basa en los posibles 8 movimientos del caballo.	Darse cuenta de que hay restricciones adicionales
Nivel 3 25,00% 25,00%	Incluye un razonamiento sobre no salirse del tablero y no volver a los ya visitados.	

Tabla 8. Niveles de respuesta y desafíos a pregunta 4

Nivel	Tipo de respuesta	Desafío de aprendizaje para subir de nivel
Nivel 0 62,50% 12,50%	No dice ninguno o solo incorrectos	Entender el problema y alinearlos con temas anteriores
Nivel 1 0,00% 0,00%	Dice algunos que habría que adaptar y ninguno correcto	Darse cuenta de la complejidad de este problema y de cómo algunos algoritmos previos no funcionarán
Nivel 2 37,50% 87,50%	Dice alguno correcto	

Tabla 9. Niveles de respuesta y desafíos a pregunta 5

Nivel	Tipo de respuesta	Desafío de aprendizaje para subir de nivel
Nivel 0 50,00% 0,00%	No responde o no presenta un algoritmo que pueda resolver el problema de la mochila	Entender el problema y la solución buscada
Nivel 1 25,00% 0,00%	Aplica una técnica anterior o describe solo una parte del algoritmo.	Darse cuenta de que es un problema complejo y que hay que hacer una búsqueda
Nivel 2 25,00% 62,50%	Describe una solución válida, aunque no razona sobre el hecho de tener que probar alternativas en paralelo.	Aprender a gestionar problemas con varias alternativas
Nivel 3 0,00% 37,50%	Incluye correctamente las alternativas y el hecho de tener que probarlas	



Tabla 10. Niveles de respuesta y desafíos a pregunta 6

Nivel	Tipo de respuesta	Desafío de aprendizaje para subir de nivel
Nivel 0 75,00 % 12,50 %	No responde o no muestra un razonamiento válido	Entender el problema y la solución buscada
Nivel 1 0,00 % 0,00 %	Da una descripción válida, aunque no concreta un valor	Analizar el problema e imaginarse una posible solución
Nivel 2 25,00 % 25,00 %	Da una solución válida pero no razona sobre el hecho de que puede quedarse sin capacidad restante	Entender que pueden existir restricciones que hagan que el algoritmo termine antes
Nivel 3 0,00 % 62,50 %	Incluye un razonamiento sobre la capacidad restante	

Tabla 11. Niveles de respuesta y desafíos a pregunta 7

Nivel	Tipo de respuesta	Desafío de aprendizaje para subir de nivel
Nivel 0 87,50 % 37,50 %	No responde o no presenta un razonamiento válido	Entender que hay acciones que no tienen sentido
Nivel 1 12,50 % 50,00 %	Razona sobre los beneficios de usar memoria o heurísticas de ordenación	Darse cuenta de que, además de tener sentido ordenar, puede no tener sentido el hecho de probar una alternativa
Nivel 2 0,00 % 12,50 %	Concluye que hay que filtrar ramas	Ser capaz de escribir un filtro admisible
Nivel 3 0,00 % 0,00 %	Incluye una propuesta concreta de un filtro	

Tabla 12. Niveles de respuesta y desafíos a pregunta 8

Nivel	Tipo de respuesta	Desafío de aprendizaje para subir de nivel
Nivel 0 87,50 % 50,00 %	No hay respuesta o no presenta un razonamiento válido	Darse cuenta de que hay muchas soluciones y que, para obtener la mejor, antes se han de haber calculado muchas
Nivel 1 12,50 % 50,00 %	Describe una propuesta voraz básica	Entender que lo mejor es encontrar una solución buena lo antes posible
Nivel 2 0,00 % 0,00 %	Describe una propuesta voraz ordenando las alternativas en base a una heurística	



De los 17 alumnos que respondieron los cuestionarios, solo 8 respondieron a los dos. Esto se pudo identificar gracias a que se le pidió un pseudónimo que utilizaron en ambos casos. Es por ello por lo que algunos niveles aparecen con 0% tanto en el inicial como final, ya que los alumnos que no respondieron a ambos han sido filtrados. En la figura 3 puede observarse un análisis individualizado por alumno de cada una de las preguntas.

		Cuestionario Inicial								
		P1	P2	P3	P4	P5	P6	P7	P8	media
alumno	1	1	0	2	2	0	0	0	0	0,75
	2	1	3	3	0	1	0	0	0	1
	3	1	2	2	0	1	0	1	0	0,875
	4	3	2	3	2	2	2	0	0	1,875
	5	2	2	2	0	0	0	0	0	0,75
	6	0	3	0	0	0	0	0	0	0,375
	7	3	3	2	0	2	2	0	1	1,625
	8	2	0	0	2	0	0	0	0	0,625
<b>media</b>		<b>1,63</b>	<b>1,88</b>	<b>1,75</b>	<b>0,75</b>	<b>0,75</b>	<b>0,50</b>	<b>0,13</b>	<b>0,13</b>	<b>0,984</b>

		Cuestionario Final								
		P1	P2	P3	P4	P5	P6	P7	P8	media
alumno	1	4	2	2	2	3	2	2	1	2,38
	2	3	2	2	2	2	3	0	1	2,00
	3	3	0	0	0	2	3	1	1	1,25
	4	4	2	0	2	3	2	0	0	1,75
	5	4	2	2	2	3	3	1	1	2,38
	6	2	0	2	2	2	0	1	0	1,25
	7	3	2	3	2	2	3	0	0	2,00
	8	2	2	3	2	2	3	1	0	2,00
<b>media</b>		<b>3,13</b>	<b>1,50</b>	<b>1,75</b>	<b>1,75</b>	<b>2,38</b>	<b>2,38</b>	<b>0,75</b>	<b>0,50</b>	<b>1,88</b>

Figura 3. Mapa de niveles inicial y final de los alumnos que respondieron ambos cuestionarios.

Aunque se observa una mejoría sustancial general en el aprendizaje del alumno, también pueden identificarse algunos casos en los que parece haberse empeorado. Analizando las respuestas, vemos que algunos alumnos han respondido vagamente a la pregunta en el cuestionario final, no incluyendo detalles que sí habían incluido en el inicial. Por ejemplo, el alumno 2 responde inicialmente a la pregunta 2: «Daría (siempre que tenga solución), el número de casillas del tablero -1» que es una respuesta

de nivel 3. Sin embargo, esta es su respuesta en el cuestionario final: «Daría 11 pasos, tantos como casillas tenga el tablero quitando la inicial» que es una respuesta de nivel 2 (cf. tabla 5).

Observando las respuestas a las preguntas 7 y 8, se ve como el nivel superior, el 2, es raramente alcanzado. Esto es debido a que estas preguntas, en sí, son de carácter avanzado y no todos los alumnos llegan aún a entenderlo al completo. El aprendizaje ocurre, no obstante, al verse incrementado el número de alumnos en el nivel intermedio, el 1.

En general, puede decirse que se observa un claro aprendizaje ya que los niveles bajos se ven reducidos en beneficio de los niveles altos, es decir, los alumnos están ahora en una escala de aprendizaje superior al que estaban inicialmente. Observándose los valores medios, también puede concluirse que, salvo pequeñas excepciones:

1. Las 8 preguntas demuestran que se ha producido un incremento en el aprendizaje de la clase en general.
2. Las respuestas de los alumnos individualmente demuestran que en todos se ha producido aprendizaje.

### Evaluación del CIMA

Personalmente, a nivel global, este CIMA lo considero positivo. Al ser más largo ha permitido estructurar los contenidos de una manera más estratégica. Aunque mi sensación es que se han dado menos contenidos que con el modelo transmisivo tradicional, aunque sí que sé que los alumnos los han asimilado mejor. Por otro lado, me ha servido para conocer que los alumnos nos tenían los conocimientos que los profesores de la asignatura damos por hecho que tienen, por lo que no están del todo preparados para recibir lo que se les planteaba dar.

Con respecto a las debilidades de este CIMA, aunque se había previsto tener cierto margen de tiempo en cada sesión, este ha resultado insuficiente. Las actividades no estaban del todo bien dimensionadas y me he visto obligado a partir en varias sesiones lo que, *a priori*, estaba diseñado para una. No obstante, esto ha supuesto alguna sorpresa positiva en algún caso ya que los alumnos mantenían la «tarea» fuera del aula.

Como mejora del CIMA planteo:

- Ampliar los tiempos de debate tras las actividades de ideación integrándolas con las actividades de conclusiones, ya que muchas de las cosas que se trataban en las conclusiones tenían más sentido tratarlas directamente cuando se discutían en el debate.
- Añadir actividades que puedan ser realizadas fuera del tiempo del aula con el objetivo de que el alumno llegue preparado a clase.



- Buscar lecturas o extractos de lecturas que los alumnos puedan procesar durante o antes de la sesión como fuente externa.
- Buscar recursos de imagen o vídeo que permitan al alumno ver de una manera gráfica los estados y evolución de los algoritmos que se enseñan en este bloque de la asignatura.

## Referencias bibliográficas

- Bain, K. (2007). *Lo que hacen los mejores profesores universitarios*. Valencia: Universitat de València.
- Finkel, D. (2008). *Dar clase con la boca cerrada*. Valencia: Universitat de València.
- Kennedy, J.O.S. (1986) Introduction to Dynamic Programming. En: *Dynamic Programming* (pp. 27-49). Dordrecht: Springer.
- Navarro Medina, E. y Porlán, R. (2020). *Ciclos de mejora en el aula. Año 2019. Experiencias de innovación docente de la Universidad de Sevilla*. Sevilla: Editorial de la Universidad de Sevilla. <https://doi.org/10.12795/9788447221912>.
- Rivero, A. y Porlán, R. (2017) La evaluación en la enseñanza universitaria. En R. Porlán (Coord.), *Enseñanza universitaria. Cómo mejorarla* (pp. 73-91). Madrid: Ediciones Morata.
- Golomb, S.W. y Baumert, L.D. (1965). Backtrack Programming. *Journal of ACM*, 12 (4), 516-524. <https://doi.org/10.1145/321296.321300>.
- Vygotsky, L. (1989). *El desarrollo de los procesos psicológicos superiores*. Barcelona: Crítica.

