

Mejora de la comprensión de las estructuras de datos

Raquel Lacuesta, Karmelo Urzelai
Departamento de Informática e Ingeniería de Sistemas
Escuela Universitaria Politécnica de Teruel
Universidad de Zaragoza
lacuesta@unizar.es, karmelo@unizar.es

Resumen

El artículo presenta el desarrollo de animaciones interactivas para dar soporte a la docencia de la asignatura Estructuras de Datos, en una ingeniería técnica en informática.

La asignatura consta de un amplio temario que hace que al alumno no le resulte fácil asimilar todos los modelos; el objetivo propuesto es facilitar su aprendizaje, así como animarle al autoestudio, poniendo en internet animaciones de soporte para su consulta.

Las animaciones desarrolladas se han clasificado en 8 grupos, cada uno de ellos asociado a un tema de la asignatura: costes computacionales, pilas, colas, listas, tablas, árboles, grafos y skip-lists. Para cada uno de estos grupos se describen las animaciones realizadas, mostrándose imágenes de algunas de ellas.

Se presenta un enfoque didáctico centrado en la motivación y refuerzo mediante animaciones interactivas que ayuden a la comprensión de la asignatura.

1. Introducción

La asignatura de Estructuras de Datos se imparte en Teruel en el primer curso de la Ingeniería Técnica en Informática de Gestión. Los alumnos llegan a la asignatura únicamente con los conocimientos adquiridos en la asignatura de Programación I impartida en el primer cuatrimestre.

Los resultados de los primeros años de implantación de la titulación han sido poco satisfactorios debido al gran número de abandonos y a al escasa número de aprobados finales.

Analizando el problema, y dejando de lado motivos ajenos a la propia asignatura y su

docencia en los que no se puede actuar, o en los que la resolución incumbe a otros entornos, las causas resultantes se han concentrado en dos aspectos: la amplitud del temario, y el todavía escaso nivel de programación y consecuente falta de capacidad de abstracción.

Analizando ambos aspectos, el temario, a pesar de ser amplio, aparte de ser coherente con la misma asignatura dada en otros centros, se considera adecuado a su número de créditos. De todas formas se ha realizado una adecuación en el tiempo dedicado a cada tema, quitando peso a algunos temas menos importantes, y dando más otros.

Por otro lado el nivel ofrecido en la asignatura previa de programación se considera correcto, y al menos de momento no se ha tomado en consideración desplazar la asignatura en el plan de estudios para dar una asignatura adicional intermedia.

2. Descripción de la asignatura.

La asignatura se divide en nueve temas, de los cuales cinco se dedican a la presentación de diferentes estructuras de datos, y donde para cada una de ellas se muestran varias implementaciones.

Los Tipos Abstractos de Datos (TAD) mostrados son: estructuras de datos lineales (pilas, colas, listas), tabla (*hashing*), árboles, grafos y *skip-lists*. [2][8]. Todos los conceptos teóricos se refuerzan en prácticas de laboratorio, que se desarrollan en el lenguaje de programación Ada 95 [1], donde deben implementarse variantes de los diferentes TADs explicados en las clases de teoría.

Lo que se pretende a lo largo del temario impartido es facilitar un esquema lógico para manipular los datos en función del problema que se deba tratar y el algoritmo que se utilice. Es importante escoger la estructura de datos y su

implementación más adecuadas, por lo que es fundamental conocerlas y comprenderlas.

Cabe destacar que se tratan tanto estructuras de datos estáticas (tamaño en memoria fijo) como dinámicas (tamaño en memoria variable), y que el planteamiento a la hora de explicarlas se debe adaptar a sus peculiaridades.

3. Motivación

El objetivo principal para la mejora de resultados, dentro del ámbito docente de la asignatura, ha sido intentar mejorar el entendimiento por parte de los alumnos de los diversos temas, intentado facilitarles la visualización de las diferentes estructuras de datos, y hacerles más ameno y llevadero el temario.

La explicación de algunas estructuras de datos implica mostrar al alumno cómo se mueven los datos, cómo cambian ciertos valores, como evolucionan por ejemplo los punteros, y todo ello es difícil hacer con medios estáticos como es la pizarra o las transparencias, que cómo mucho pueden mostrar imágenes inanimadas.

Se han desarrollado ya diversas aplicaciones para mejorar la comprensión de algunas estructuras [7], queriendo aquí ampliar las posibilidades de una forma muy visual.

Por tanto tras considerarse insuficientes los medios utilizados hasta ahora, se decidió incorporar las nuevas tecnologías para mostrar la evolución de las estructuras de datos de forma animada. A su vez se consideró que una vez que se realizaba el esfuerzo sería interesante que estas animaciones no se vieran de forma pasiva sino que sería conveniente la implicación del alumno, haciendo que éstas fueran interactivas exigiendo al menos un mínimo de actuación por su parte, y en algunos casos añadiendo ventanas explicativas de los pasos seguidos.

La interactividad de las animaciones facilita un mejor entendimiento de los conceptos, frente a aquellas animaciones que sólo proporcionan una visualización del dinamismo de la estructura, ya que entre otras cosas permiten al alumno seguir la animación paso a paso de acuerdo a sus propias necesidades, y a su velocidad de asimilación.

Las animaciones han sido desarrolladas siguiendo el proceso de aprendizaje de un alumno que desconocía la asignatura, para plasmar desde una perspectiva más realista las dificultades encontradas y expresar de forma más cercana al

alumno las ideas [6]. Lógicamente en todo momento el profesor supervisaba la corrección de las operaciones mostradas en las animaciones y su aspecto pedagógico. Así mismo es el profesor quien con su experiencia ha escogido las animaciones más representativas y las estructuras de datos e implementaciones que más dificultad presentan habitualmente para su aprendizaje.

Por otro lado se ha ajustado el conjunto de animaciones a las estudiadas dentro de la asignatura, abarcando todo el temario. Y no sólo se han desarrollado animaciones pensando en el comportamiento del T.A.D., sino que en algunos casos la animación se ha centrado más en una posible implementación, que es la que más dificultad de comprensión tenía. Y todo ello se ha realizado buscando siempre la perspectiva más didáctica posible.

El empleo de animaciones se ha utilizado también en otras materias, por ejemplo para la animación y simulación de algoritmos paralelos de exploración de grafos[3]. También se han hecho animaciones para representar estructuras de datos por ejemplo en [7], no adecuándose a la asignatura Estructuras de Datos sino a la de Programación, por lo que resultan insuficientes para impartir un temario completo de la asignatura Estructuras de Datos.

4. Objetivos

Con esta iniciativa nos proponemos:

- Ofrecer al alumno un material de apoyo que facilite el aprendizaje. Todas las animaciones están puestas a disposición de los alumnos para repasar cuantas veces necesite el temario impartido. Cada animación irá adjunta al tema correspondiente.
- Incentivar al estudiante mediante apoyos visuales a preparar la asignatura. La unión de los temas teóricos a las animaciones gráficas facilita y ameniza el aprendizaje.
- Promover el uso de nuevas tecnologías por parte del alumno. Tanto los temas de la asignatura como las animaciones están en la página *web* de la asignatura con lo que el alumno puede acceder fácilmente a todo el material, tanto desde casa como desde la universidad
- Mejorar la comprensión de conceptos durante la clase. El profesor puede utilizar las diversas animaciones durante las clases para hacer

comprender a los alumnos los temas explicados.

Como conclusión, la base principal para la mejora docente está en el apoyo que las animaciones pueden dar tanto durante las clases de teoría y prácticas, como durante la fase de autoaprendizaje que posteriormente el alumno lleve a cabo para asentar los conceptos explicados previamente.

Y por otro lado las animaciones son un gran apoyo para todas aquellas personas que no pudiendo asistir a las clases presenciales desean seguir la asignatura, ya que muchas veces los apuntes de otros alumnos resultan totalmente insuficientes para seguir una explicación oral que mostraba el flujo de datos dentro una estructura.

5. Animaciones

El entorno utilizado para el desarrollo de las animaciones ha sido Flash [4][5] debido a su facilidad de uso en la creación de películas dinámicas e interactivas, consiguiendo, además, una optimización en el tamaño de cada animación mediante la utilización de gráficos vectoriales.

Para la visualización de las animaciones desde las páginas *web* de la asignatura será necesario el programa *Flash Player*, gratuito, de fácil acceso y disponible en casi todos los navegadores.

Cada estructura de datos tiene su correspondiente animación, y algunas de ellas disponen de varias animaciones ya sea por su dificultad, o por la variedad de implementaciones. Dentro de cada animación se podrán realizar las operaciones pertinentes a cada estructura, consiguiendo así la interactividad del alumno con la animación y por lo tanto un refuerzo personalizado.

Para cada una de ellas se estudiaron las diferentes posibilidades de representación, escogiéndose la más intuitiva. El objetivo perseguido fue el de conseguir un efecto de autoaprendizaje y una fácil asimilación de los contenidos durante su visualización.

Dentro de las animaciones hay algunas cuyo objetivo es mostrar de forma detallada el comportamiento de la estructura y otras cuya única función es dar al estudiante la idea intuitiva de la estructura. Por ejemplo en la explicación de las colas una primera animación representa el comportamiento de una cola de coches en una

parada, pudiendo añadir coches y quitar coches, lo que ya de por sí puede dar la idea del comportamiento de una cola mejor que ciertas explicaciones formales. Una segunda animación representa ya una secuencia de datos en los que podemos realizar las operaciones de *encolar* y *desencolar*.



Figura 1. Concepto de cola

Todas las animaciones disponen de una serie de botones que permiten, dentro de ciertas restricciones, decidir la evolución de los datos, ver la animación paso a paso, reiniciar, o elegir entre diversas opciones. Algunas de ellas además disponen de cuadros de texto en los que se va explicando paso a paso las acciones que se están realizando y la fase en la que se encuentra.

En las animaciones que trabajan directamente con datos se ha mantenido una interfaz uniforme para darle homogeneidad, facilitar su seguimiento y no distraer al alumno.

Las animaciones realizadas [6] han sido:

1. Costes computacionales: La problemática de este tema es por un lado que su formalismo hace que el alumno sea más reacio a su estudio y por otro considerar que la eficiencia es un tema menor a causa de la creciente potencia de los ordenadores. La animación desarrollada por tanto incide en ambos aspectos de modo que por un lado presenta gráficamente los diferentes tipos de costes, y a continuación de una forma visual muestra las grandes diferencias entre unas y otras, para que el alumno sea consciente de la importancia de elegir un algoritmo de un coste razonable.
2. Pílas: Las animaciones de estructuras comienzan con ésta primera, muy sencilla, y

que introduce al alumno en la utilización de las animaciones que encontrará más tarde, que serán más complejas y elaboradas. En este caso una primera animación ofrece la idea intuitiva de pila por medio del apilamiento y desapilamiento de unos cubos, y otra animación posterior ya muestra el comportamiento más formal de una estructura de datos *pila*, con sus operaciones de *apilar*, *desapilar*, *cima* y *vacía*. Esta segunda animación ya introduce al alumno en la estética general de las animaciones.

3. Colas: A pesar de ser también una estructura muy sencilla se han definido dos animaciones, al igual que en las pilas una primera para aportar la idea intuitiva de *cola* (Figura 1), y una segunda para la definición más formal. Se intenta así con estas primeras lecciones atraer al alumno a “jugar” con las animaciones, de modo que posteriormente con animaciones más complejas le resulte más natural.

4. Listas con punto de interés: Para este caso, siguiendo la línea de las dos anteriores, se ha creado una primera animación para entender el concepto por medio de un pequeño editor de texto simulado. Y por otro se han construido una serie de animaciones orientadas a mostrar el comportamiento de una *lista* implementada de forma encadenada. Pese a explicarse en el temario también otro tipo de implementaciones se ha considerado innecesario realizar animaciones para ellas debido a su simplicidad.

De las animaciones realizadas se ha incidido en los aspectos más críticos como la creación de la lista, para que se entiendan los conceptos de apuntador y de pila de posiciones libres. Otras de las animaciones muestran las operaciones más habituales de las listas y que pueden resultar difíciles de seguir sólo con una explicación oral, como el avance del *Punto de Interés* (Figura 2), la inserción de un nuevo elemento o el borrado. Las animaciones avanzan paso a paso, bajo el control del alumno, para que vea los cambios producidos tanto en el encadenamiento de la lista como en la gestión del espacio libre.

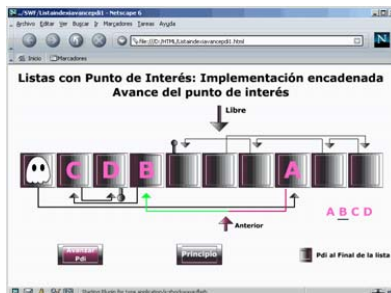


Figura 2. Listas con Punto de Interés

5. Tablas *Hash*: Parte de la dificultad de esta estructura de datos es mostrar el comportamiento de los tres tipos básicos de implementaciones que pueden darse a las tablas de dispersión: encadenamiento indirecto, encadenamiento directo y no encadenadas. Por ello se han realizado animaciones que muestran cómo se pueden implementar cada una de ellas y su comportamiento, mostrándose por ejemplo para las primeras, en tres animaciones, la inserción de un elemento en la tabla, el borrado de elementos y la consulta de si un elemento pertenece a la tabla (Figura 3).



Figura 3. Búsqueda en tabla *hash*

6. Árboles: En el tema de árboles a modo de ejemplo se ha realizado una animación, que llama a otras tres animaciones, que muestran los tres tipos de recorrido en profundidad (Figura 4), y otra animación para mostrar el recorrido en anchura. A causa de que la implementación del recorrido en anchura de los árboles es un poco más complejo por la necesidad de una cola

que permita visitar los nodos en el orden correcto, en la animación a medida que se recorre el árbol se muestra también paralelamente paso a paso la evolución de la cola que nos permite realizar dicho recorrido

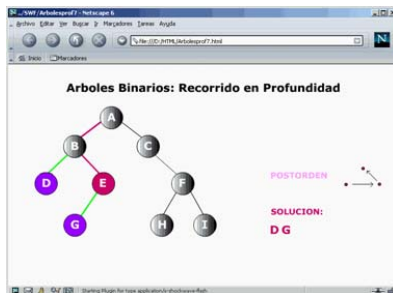


Figura 4. Recorrido en árboles

7. Grafos: Como ejemplo de utilización de grafos se ha realizado una animación que muestra la ejecución del algoritmo de Dijkstra. En dicha animación se avanza poco a poco seleccionando el siguiente nodo del camino de coste mínimo, y calculando los costes de los caminos a los demás nodos. Se muestra de forma explícita la tabla de costes y los caminos que los generaron, al mismo tiempo que se representa el grafo y se van marcando dinámicamente los diferentes caminos que se están calculando.

8. Skip List: Para mostrar esta estructura de datos, se ha dibujado una *skip-list* sobre la que se ha realizado el proceso de inserción de un elemento (Figura 5). Para la inserción en primer lugar se hace la búsqueda del punto de inserción, con lo que se está mostrando también cómo funciona esta operación, pero para la posterior inserción además se van guardando los enlaces para poder redireccionarlos después.

6. Ejemplo de Animación

A modo de ejemplo se muestra paso a paso una de las animaciones asociadas a la estructura más simple de las realizadas: la pila.

Como ya se ha dicho, hay una primera animación que muestra la idea intuitiva de pila por

medio de unos cubos etiquetados con letras, y la que aquí se explica es la segunda, que muestra, con la estética general de las animaciones, el comportamiento del TAD pila y sus operaciones.

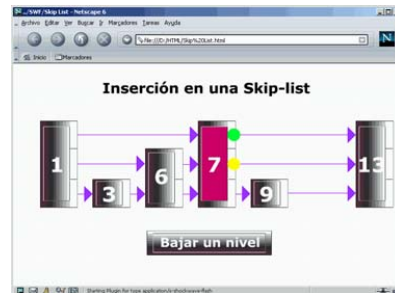


Figura 5. Skip-List

La animación comienza con una pila vacía, y el indicador *Pila Vacía* parpadeando para mostrar que está activado. El marcador de la cima de la pila está vacío al no haber ningún elemento en ella (Figura 6).



Figura 6. Pila vacía

Si se pulsa el botón de *Apilar*, el primer dato aparece por la izquierda y se desplaza hasta entrar en la pila y colocarse como cima. El indicador de *Pila Vacía* se desactiva y el dato introducido pasa a mostrarse en la ventana *Cima de Pila* (Figura 7).



Figura 7. Apilado de un dato

Si se pulsa *Apilar* más veces el proceso se repite llenándose poco a poco la pila, y modificándose el marcador de la cima de la pila (Figura 8).



Figura 8. Apilado de tres datos

En cualquier momento se puede pulsar también el botón de *Desapilar*, en cuyo caso el elemento de la cima, sale de la pila y desaparece por la izquierda (Figura 9).



Figura 9. Desapilando un dato

En el caso de desapilarse todos los elementos vuelve a encenderse el indicador de *Pila Vacía*.

El resto de animaciones pueden verse en <http://eupt.unizar.es/eda/temario/Temario.html>¹

7. Conclusiones

Ante la problemática planteada en la asignatura Estructuras de Datos en la que los alumnos abandonaban fácilmente a causa de una temática amplia y la introducción de algunos conceptos nuevos, se ha decidido apoyar tanto las clases como el autoestudio con animaciones.

Las animaciones se han realizado en Flash por su versatilidad, facilidad de uso y por ser prácticamente un estándar de *Internet*, ya que uno de los destinos principales de dichas animaciones es colocarlas en las páginas web de la asignatura, de modo que los alumnos puedan consultarlas en cualquier momento. También podrán ser utilizadas durante las clases para las explicaciones.

Las animaciones han sido realizadas por una persona que desconocía previamente la asignatura de modo que pudiera plasmar sus propias dudas y problemas durante el aprendizaje en las mismas animaciones. Lógicamente dichas animaciones en todo momento han estado guiadas por el profesor de la asignatura que además de aportar la experiencia sobre las dudas más habituales de los alumnos y controlar la corrección de todo lo realizado, volcaba su experiencia docente en la búsqueda de la representación más pedagógica posible

A lo largo del segundo cuatrimestre del curso 2002-2003 se van a utilizar las animaciones desarrolladas como herramienta docente, esperando una buena recepción del esfuerzo desarrollado, siendo positiva la impresión de los alumnos que ya las han visto. De todas formas con la realimentación de la reacción y la opinión de los alumnos se espera mejorar, adecuar y ampliar este primer paso.

Por último, consideramos que con este desarrollo no se completa ningún trabajo, sino al contrario la experiencia nos ha mostrado que se abren las puertas para la utilización de animaciones en muchas otras asignaturas, en áreas que pueden ser completamente diferentes.

¹ Prohibida la reproducción, copia o utilización de cualquier información de la web, sin la expresa autorización de su autor

Agradecimientos

Los autores de este artículo desean agradecer a Celia Marzo su dedicación y desarrollo de las animaciones. También desean agradecer a Elvira Mayordomo, del mismo Dpto., la iniciativa y desarrollo del proyecto, llevando ella posteriormente una línea totalmente independiente a pesar de compartir objetivos.

Referencias

- [1] Barnes J. "Programming in Ada 95". Addison Wesley. 1999
- [2] Franch X. "Estructuras de datos. Especificación, diseño e implementación". Edición UPC.
- [3] José Fco. Cachairo González, Manuel Díaz Rodríguez, Antonio Vallecillo Moreno . "Animación y Simulación de Algoritmos Paralelos de Exploración de Grafos".
- [4] Macromedia. "Manual Macromedia Flash 5. Guía de Consulta de Actionscript". Macromedia Inc.
- [5] Macromedia . "Manual Macromedia Flash 5. Utilización de Flash". Macromedia Inc.
- [6] Mazo C. "Animaciones Flash para la asignatura Estructura de Datos". Trabajo Fin de Carrera. Escuela Universitaria Politécnica de Teruel.
- [7] Salamó M., Camps J., Vallespi C., Vernet D., Llorá X., Bernadó E., Garrell J.M., González X.. "Iniciativas para motivar a los alumnos de Programación". Universitat Ramon Llull.
- [8] Weiss M.A. "Estructuras de datos y algoritmos". Addison-Wesley-Iberoamericana.