

Informe final de ejecución del Proyecto de Innovación y Mejora Docente realizado durante el curso académico 2016-2017 en el Departamento de Informática y Automática de la Facultad de Ciencias, financiado mediante la convocatoria de 2016 del Programa de Mejora de la Calidad- Plan Estratégico General 2013-2018.

ID2016/184

Entorno Virtual para la docencia on-line de Estructuras de Datos

Miembros del Equipo de Trabajo:

María José Polo Martín

Ángeles María Moreno Montero

Iván Álvarez Navia

Susana Álvarez Rosado

Belén Curto Diego

Vidal Moreno Rodilla

Francisco Javier Blanco Rodríguez

Departamento de Informática y Automática.
Facultad de Ciencias de la Universidad de Salamanca.
Plaza de los Caídos S/N, 37008

El objetivo principal del proyecto es la creación de una aplicación web para la visualización del comportamiento de diferentes algoritmos sobre varias estructuras de datos, principalmente árboles, que se estudian en la asignatura Estructuras de Datos y Algoritmos II. Esta asignatura se enmarcan dentro de la materia de Programación en el Grado de Ingeniería Informática. Complementan la formación obtenida en las asignaturas de primer curso y proporcionan una base para asignaturas posteriores, como Redes de Computadores.

Introducción

En este proyecto se ha diseñado y desarrollado una aplicación web para la visualización on-line del comportamiento de diferentes algoritmos sobre varias estructuras de datos. Se han introducido algunas mejoras al software educativo que se desarrolló dentro del proyecto Herramienta para el aprendizaje visual y dinámico de estructuras de datos [9]. Esta herramienta consiste en una aplicación de escritorio que el estudiante tiene que descargarse e instalar en su ordenador. La mejora principal ha consistido en convertir esta herramienta en una aplicación web. De esta forma se aumenta la disponibilidad, accesibilidad y facilidad de uso a los contenidos prácticos que cubre la aplicación. Además se han ampliado algunas funcionalidades que se habían echado en falta en la aplicación de escritorio anterior. La ampliación de estas funcionalidades surgen como resultado de los guiones de prácticas elaborados en el desarrollo del proyecto Creación de material para prácticas de estructuras de Datos Complejas [10].

Contexto de desarrollo

En la asignatura de Estructuras de Datos y Algoritmo II se presentan al estudiante gran cantidad de algoritmos que manejan estructuras de datos, [7], [2], [1], [8], como árboles y grafos. Su aprendizaje plantea ciertas dificultades a los estudiantes de Ingeniería Informática. Con la herramienta de escritorio utilizada en cursos previos se ha facilitado este aprendizaje, pues permite aplicar los algoritmos a nuevos ejemplos, comprobar el resultado de casos más complejos y ver de forma dinámica la evolución del algoritmo sobre un ejemplo concreto [9].

En las sesiones prácticas se pasa del diseño y análisis de los algoritmos, a su implementación en un lenguaje de programación y, a su posterior ejecución y prueba. Para ello el estudiante dispone de una serie de guiones de prácticas desarrollados en el proyecto Creación de material para prácticas de estructuras de Datos Complejas [10]. Estos guiones sirven al estudiante como directriz y le permiten una comprensión escalonada de los conceptos estudiados en teoría. A los guiones de prácticas se ha ido añadiendo material de apoyo para su realización, principalmente código fuente para que las prácticas se centren en la implementación de los algoritmos estudiados. En este sentido hemos considerado interesante que la aplicación fuera capaz de generar el código fuente de los ejemplos que muestra de manera visual. De esta forma, una vez implementado el algoritmo, se dispone de los mismos ejemplos y es posible concluir con las fases de ejecución y prueba. La idea es poder aplicar su propio algoritmo al mismo ejemplo de prueba generado por la aplicación visual y, de esta forma, constatar el resultado veraz o no de su implementación. Se plantea una nueva funcionalidad que no estaba diseñada en la aplicación de escritorio, la necesidad de obtener en código C los mismos ejemplos que la herramienta visual muestra.

Además, la herramienta visual de escritorio permite visualizar la ejecución del algoritmo paso a paso sobre el ejemplo. Para ayudar a la comprensión de los algoritmos, se plantea la posibilidad no solo de mostrar el propio algoritmo, sino que la ejecución paso a paso se muestre sobre él. Esta nueva funcionalidad ayudará a la comprensión de los algoritmos y resulta especialmente importante en la ejecución de los algoritmos recursivos que, aún siendo sencillos de implementar, presentan muchas dificultades a los estudiantes en el seguimiento de su ejecución.

Objetivos del proyecto

En la fase inicial del proyecto quedaron definidos los objetivos del mismo. El objetivo principal ha sido la conversión de una aplicación de escritorio en una aplicación web, de forma que puede utilizarse a través de cualquier navegador que permita acceso a Internet, es decir, ya no es necesario descargar ni instalar ningún tipo de archivo en nuestro ordenador para trabajar con la aplicación. La mayoría de objetivos ya estaban definidos en el proyecto de partida, en esta etapa, por tanto, se ha definido la mejora del proyecto y las posibles ampliaciones.

Como objetivos comunes de la aplicación de escritorio y la aplicación web, rescatamos los objetivos que se plantearon en la aplicación de escritorio. Por cada estructura estudiada la herramienta debía permitir la realización de diferentes actividades:

- Estudio de ejemplos proporcionados por el profesor
- Posibilidad de generar ejemplos específicos por parte del estudiante y ejemplos aleatorios automáticos
- Posibilidad de guardar, modificar y recuperar los ejemplos generados para su posterior visualización y estudio
- Análisis por parte del estudiante de los ejemplos generados mediante la ejecución paso a paso de todos los algoritmos estudiados en clase
- Verificación de la comprensión de los algoritmos aplicados a los ejemplos: el resultado final que proporcione la aplicación es correcto y permite verificar los resultados esperados por el estudiante.

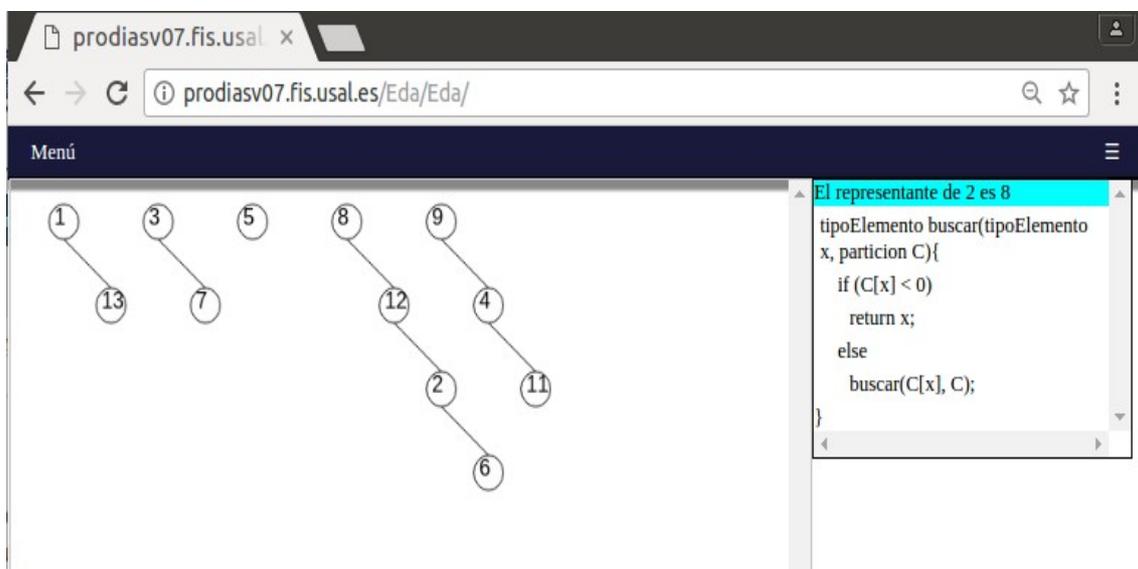
En esta fase, desarrollada durante los meses de Septiembre y Octubre de 2016, se decidió ampliar algunas funcionalidades visuales respecto a la versión de escritorio. En este sentido se añadieron dos funcionalidades nuevas:

- Posibilidad de añadir a la ejecución paso a paso el seguimiento no solo en el ejemplo gráfico sino también en la línea de ejecución del algoritmo en ese momento

- Posibilidad de generar los fichero en lenguaje de programación C para que el estudiante pueda aplicar los algoritmos que implementa en prácticas a los ejemplo que visualiza en la propia aplicación

Descripción de la Aplicación

En el análisis, diseño y desarrollo de la aplicación, fases 2 y 3, se ha seguido el Proceso Unificado[4] y se ha utilizado el lenguaje de modelado UML. Se ha redefinido completamente la interfaz del usuario, pero sigue mostrando claramente las diferentes opciones en cuanto a tipo de estructura, algoritmo a presentar, tipo de ejecución, etc.



- figura 1 -

En la figura 1 pueden observarse las tres partes principales en que está dividida la interfaz: parte superior o menú, ventana principal a la izquierda en al que se mostrará el ejemplo de la estructura elegida y ventana secundaria a la derecha en la que se mostrará visualmente el algoritmo.



- figura

2-

La figura 2 muestra el menú principal desplegado en sus diferentes opciones. Sin embargo, dependiendo del dispositivo y del zoom elegido en el navegador, la interfaz de la aplicación puede adaptarse para una mejor visibilidad. La figura 2 muestra una posible adaptación que se genera de forma automática al aumentar el zoom en el navegador. Si el zoom disminuye las diferentes opciones del menú se muestran en la zona superior, donde van apareciendo diferentes líneas de menú, según las opciones elegidas (figura 3).

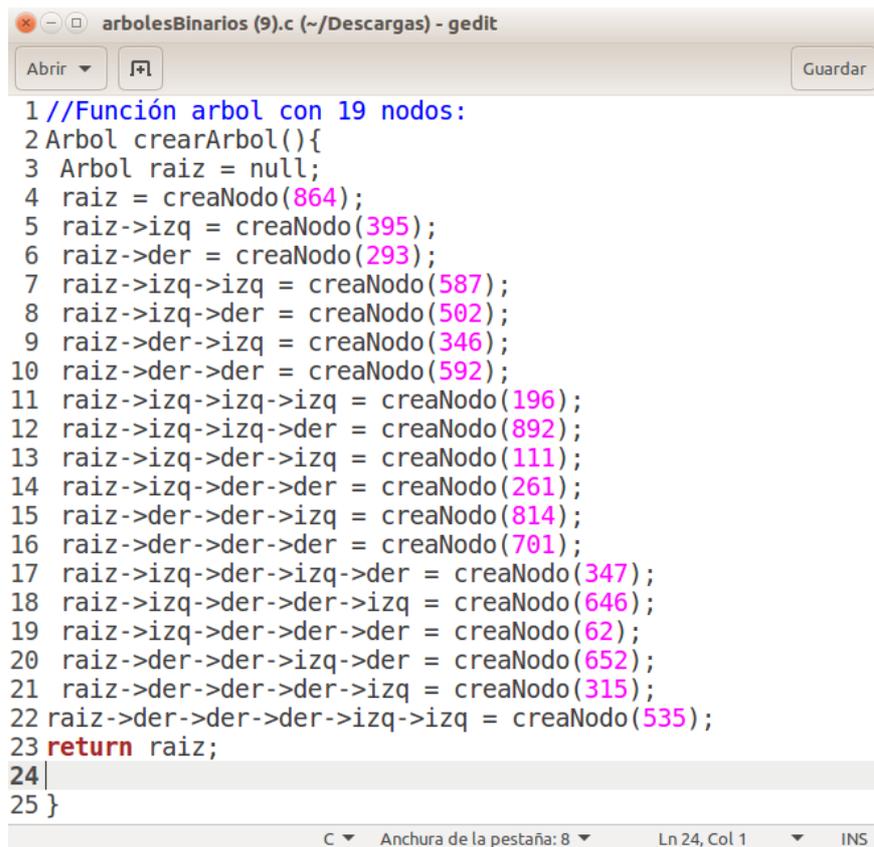
Este menú permite acceder de forma rápida y sencilla a las diferentes estructuras de datos que maneja la aplicación. Una vez elegida la estructura a estudiar, aparecen otras líneas de menús como muestra la figura 3 y, dependiendo de las opciones elegidas van apareciendo las estructuras y algoritmos en las ventanas principal y secundaria de la aplicación. En este caso se ha elegido la estructura Árboles Binarios en el menú principal, se ha generado un árbol aleatorio eligiendo esta opción en el menú intermedio. El árbol aleatorio generado puede verse en la ventana principal y el algoritmo aplicado en la ventana secundaria. El menú inferior permite seleccionar el modo de ejecución y también crear los ficheros que guarden la simulación y generen el código C que permita crear el árbol generado aleatoriamente en la ventana principal. En la parte de la derecha se muestra el algoritmo aplicado a la estructura y si se selecciona la ejecución paso a paso, va resaltando la línea concreta que se va ejecutando en cada momento.

The screenshot shows a web browser window with the URL 'prodiasv07.fis.usal.es/Eda/Eda/'. The navigation menu includes 'Arboles binarios', 'Arboles binarios de búsqueda', 'Arboles balanceados', 'Montículos binarios', 'Conjuntos disjuntos', and 'Grafos'. Below the menu are buttons for 'Paso a paso', 'Descargar archivo .c', and 'Descargar simulación'. The main area displays a binary tree with nodes containing numbers. A code editor on the right shows a C function 'int Altura(Arbol raiz)' that calculates the height of the tree.

- figura 3 -

La interfaz ha cambiado respecto a la aplicación de escritorio pero se siguen teniendo las mismas funcionalidades. Además pueden observarse las dos funcionalidades nuevas. La posibilidad de generar código C, segundo botón del menú inferior y; el seguimiento del algoritmo, si se elige el botón paso a paso, en la parte derecha de la interfaz.

El código C correspondiente al árbol de la figura 3 puede observarse en la figura 4. Esta figura muestra que la aplicación genera una función en código C que el estudiante puede incluir a su propio código de manera fácil y que le permite trabajar con la misma estructura que visualiza en la herramienta. La inclusión puede ser copiando y pegando el código C generado en sus propios ficheros de implementación o creando un nuevo fichero que puede compilar y enlazar en su proyecto de prácticas de manera rápida y sencilla.



```
arbolesBinarios (9).c (~/Descargas) - gedit
Abrir Guardar
1 //Función arbol con 19 nodos:
2 Arbol crearArbol(){
3 Arbol raiz = null;
4 raiz = creaNodo(864);
5 raiz->izq = creaNodo(395);
6 raiz->der = creaNodo(293);
7 raiz->izq->izq = creaNodo(587);
8 raiz->izq->der = creaNodo(502);
9 raiz->der->izq = creaNodo(346);
10 raiz->der->der = creaNodo(592);
11 raiz->izq->izq->izq = creaNodo(196);
12 raiz->izq->izq->der = creaNodo(892);
13 raiz->izq->der->izq = creaNodo(111);
14 raiz->izq->der->der = creaNodo(261);
15 raiz->der->der->izq = creaNodo(814);
16 raiz->der->der->der = creaNodo(701);
17 raiz->izq->der->izq->der = creaNodo(347);
18 raiz->izq->der->der->izq = creaNodo(646);
19 raiz->izq->der->der->der = creaNodo(62);
20 raiz->der->der->izq->der = creaNodo(652);
21 raiz->der->der->der->izq = creaNodo(315);
22 raiz->der->der->der->izq->izq = creaNodo(535);
23 return raiz;
24 |
25 }
```

C Anchura de la pestaña: 8 Ln 24, Col 1 INS

- figura 4 -

Conclusiones

Se ha desarrollado una aplicación web para la visualización del comportamiento de los algoritmos que se presentan en la asignatura Estructuras de Datos y Algoritmos II. La aplicación se ha presentado en las clases prácticas de la asignatura y, aunque todavía no se ha podido probar completamente y analizar los resultados obtenidos, ha servido para detectar algunos fallos y errores que estamos intentado subsanar para el próximo curso.

La utilización de la aplicación web ha aportado algunas ventajas en el estudio de la asignatura de Estructuras de Datos y Algoritmos, sobre todo en el tema de árboles que es en el que más incidencia hemos hecho. En este sentido, podemos indicar que su uso favorece el proceso educativo, pues permite la obtención on-line de material docente. En las clases presenciales este material permite plasmar los contenidos de las mismas, intercalando de forma paulatina, algoritmos teóricos con ejemplos prácticos. Por otro lado, la aplicación está disponible para el estudio individual y no presencial del estudiante. Con ella podrá simular los algoritmos que va estudiando y aplicarlos a ejemplos concretos. Con la nueva funcionalidad añadida, los ejemplos simulados pueden obtenerse en código fuente y servir como entrada a los algoritmos reales que el alumno implementa. Es una herramienta, por tanto, que permite el autoestudio y la autoevaluación.

La aplicación web, complementa la metodología docente y el material de prácticas elaborado para esta asignatura como parte del proyecto [10]. Recordamos que una de las

mejoras que proponíamos para este proyecto era la aportación de material adicional. Principalmente la inclusión de código fuente para que el estudiante, a la hora de realizar las prácticas, pudiera centrarse en la implementación de los algoritmos que ha estudiado y analizado. La herramienta, como hemos visto, permite generar parte de este código, pues de forma automática permite obtener funciones en C que crean los ejemplos que visualmente se muestran. Esta aportación ha resultado muy interesante para los alumnos, pues permite, a través del mismo ejemplo, estudiar y analizar el algoritmo teórico y después seguir trabajando con el mismo ejemplo, en las sesiones de prácticas. Sin la herramienta el estudiante debía generar este código, lo cual restaba tiempo para dedicar a los objetivos principales de la asignatura. Por otra parte, la obtención de forma automática evita posibles errores en la codificación.

Referencias

- [1] A. V. Aho, J. E. Hopcroft, and J. Ullman. Estructuras de datos y algoritmos. Addison-Wesley Iberoamericana, 1988.
- [2] O. Cairó and S. Guardati. Estructuras de datos. McGraw-Hill, 2006.
- [3] E. W. Dijkstra. A note on two problems in connection with graphs. *Numeriche Mathematics*, 1:269-171, October 1959.
- [4] I. Jacobson, J. Rumbaugh, and G. Booch. El Proceso Unificado de desarrollo de software. Addison Wesley, 2000.
- [5] I.J. James Rumbaugh and G. Booch. El Lenguaje Unificado de Modelado. Manual de Referencia. Addison Wesley, 2000.
- [6] Polo Martín, M.J.; Moreno Montero, A.M.; Iglesias Alonso, C. Experiencias en el uso de Técnicas de Visualización en el aprendizaje de Estructuras de Datos. Primeras Jornadas de Innovación Docente en la Universidad de Salamanca, pg. 276. Noviembre, 2011.
- [7] M. Weiss. Estructuras de datos y algoritmos. Pearson Education, 1995.
- [8] N. Wirth. Algoritmos y estructuras de datos=Programas. Ediciones del Castillo, 1980.
- [9] ID11/150 Herramienta para el aprendizaje visual y dinámico de estructuras de datos
- [10] ID2012/193 Creación de material para prácticas de Estructuras de Datos Complejas