Una Herramienta para Analizar Representaciones del Tipo Abstracto Diccionario

Edelmira Pasarella Sandra Zabala *
Departamento de Computación y
Tecnología de la Información
Universidad Simón Bolívar
Tel. +58-2-9063241/42 Fax +58-2-9063243
Aptdo 89000, Caracas 1080-A
Venezuela

Resumen

En este artículo se presenta una herramienta para el análisis de las representaciones del Tipo Abstracto de Datos Diccionario (TADD).
La herramienta permite realizar las operaciones básicas del tipo para diferentes representaciones, registrando sus costos. Además, ofrece al usuario la posibilidad de obtener estadísticas de dichos costos y generar gráficos que faciliten el análisis comparativo de los resultados obtenidos, lo que contribuye a la formación de criterios que sirven como base para escoger la representación más adecuada para el tipo diccionario, dado un problema que requiera de su uso.

Palabras Claves: Tipos Abstractos de Datos, Diccionario, Representaciones, Herramienta.

*{edelmira,szabala}@usb.ve

1 Introducción

Con el fin de dotar a los estudiantes del pregrado en Ciencias de la Computacion de herramientas interactivas que le sirvan de apoyo en el proceso de aprendizaje se han desarrollado diferentes sistemas en áreas como grafos [6], programación [11], algoritmos [4, 12], etc. En este trabajo se presenta una herramienta enmarcada en el ámbito de los Tipos Abstractos de Datos.

El sistema TADD fue diseñado como un instrumento para el laboratorio propuesto en [9], donde se plantea un Enfoque Experimental para la enseñanza de Tipos Abstractos de Datos en los cursos básicos de CS, en el cual en cada sesión de laboratorio, los estudiantes realizan múltiples experimentos sobre un tipo determinado, que permiten responder las siguientes preguntas:

1. Dada una secuencia de operaciones sobre un objeto del tipo, ¿Cuál representación es más eficiente?

2. Dada una representación del tipo abstracto, ¿Para qué tipo(s) de secuencia(s) de operaciones es más eficiente?

Para responder la primera pregunta, se debe armar un conjunto de experimentos que se caractericen por diferir sólo en la representación. Para responder la segunda, se arma un conjunto de experimentos que varíe sólo en la secuencia de operaciones.

El estudiante debe, por tanto, realizar sus experimentos y analizar comparativamente los resultados para poder sacar las conclusiones requeridas.

Con el propósito de que la atención del estudiante se centre en diseñar experimentos adecuados y analizar los resultados de su realización, surgió la idea de tener *Herramientas de Experimentación* que permitan montar diversos experimentos sobre un tipo dado y generar los resultados estadísticos sobre los costos de las operaciones de los mismos.

TADD es una de estas herramientas de experimentación. El Tipo Abstracto sobre el que trabaja es el Diccionario [10]. Una de las funciones de TADD es la de permitir al usuario montar, realizar y obtener resultados de experimentos sobre el tipo Diccionario.

Otra de las funciones de la herramienta es brindar al usuario información acerca del tipo Diccionario y, además, proveer mecanismos que le permitan comprender la forma en que se realizan las operaciones definidas para éste, dada una representación. Estas son funciones didácticas y aumentan el valor de la herramienta como apoyo a la enseñanza teórica y práctica del tema.

El resto del trabajo está organizado como sigue: en la sección 2 se presenta una descripción general de TADD y en la sección 3 se dan las conclusiones.

2 Descripción de la Herramienta

TADD es una herramienta de experimentación que cumple una doble función. Por un lado, permite al estudiante familiarizarse con el Tipo Abstracto de Datos Diccionario y sus representaciones. Por otro lado, facilita el estudio comparativo de las diferentes representaciones del tipo ofreciendo información asociada a la realización de diversos experimentos.

A continuación se presentan los aspectos principales de TADD, una descripción completa se encuentra en [14].

2.1 Experimentos

Un experimento se define como una 4-upla (T, t_0, R, O), donde T es el tipo abstracto de datos en estudio, t_0 es un objeto de tipo T sobre el que se realizará el experimento (objeto inicial), R es una representación para T y O es la secuencia de operaciones a aplicar durante el experimento.

Los componentes de un experimento en la herramienta TADD son los siguientes:

- T: El valor de T está implícito (Diccionario), por lo que no hace falta definir su valor al montar un experimento.

- d_0: Inicialmente, para todo experimento $d_0 = vacío$. Sin embargo, se puede definir un d_0 diferente, con un grupo de elementos de tipo básico sobre los que se aplicará la secuencia de operaciones (en la versión actual de TADD, los elementos de un diccionario son números, que constituyen tanto la clave como la información asociada a la clave). d_0 puede ser creado interactivamente por el usuario, recuperado de memoria secundaria o generado aleatoriamente.

- O: Inicialmente la secuencia de operaciones es, por defecto, vacía. El usuario puede crearla interactivamente o generarla de manera aleatoria. Las operaciones que se pueden realizar sobre un diccionario son: Inserción, Eliminación y Búsqueda de un elemento. Esta secuencia de operaciones es llamada escenario del experimento.

- R: Inicialmente la representación es, por defecto, un arreglo. El usuario puede seleccionar cualquiera de las representaciones que ofrece TADD.

El montaje de un experimento consiste en definir los valores para las componentes de la 4-upla y la realización es la aplicación de la composición de las operaciones sobre el diccionario inicial.

Los experimentos pueden ser montados interactivamente (creación interactiva de sus componentes), generados aleatoriamente o recuperados de memoria secundaria. Además, los experimentos, al igual que los diccionarios también pueden ser salvados en archivos para su posterior utilización.

Después de montado un experimento éste puede realizarse. Todo experimento parte de un diccionario *vacío*. Si d_0 no es vacío, la realización del experimento se traduce en realizar primero operaciones de inserción en el diccionario para todos los elementos de d_0 y luego aplicar la secuencia de operaciones O.

2.2 Representaciones

Las representaciones para un diccionario ofrecidas por la herramienta son: Arreglos ordenados o no ordenados, Tablas de *Hash* (abiertas o cerradas) y Árboles (Binarios de Búsqueda o AVL). Los algoritmos generales fueron tomados de [1, 13]; en el caso de Arreglos ordenados, los algoritmos de las operaciones están basados en la búsqueda binaria y si la representación es un Arreglo no ordenado, en la búsqueda secuencial. Los algoritmos para Árboles binarios y AVL fueron tomados de [8]. En el caso de *hashing* abierto, el usuario puede escoger entre cuatro funciones de *hashing* diferentes: Método de la división, Esquema de multiplicación, Razón dorada y Aleatoria. Si se trata de *hashing* cerrado, adicionalmente se tiene que seleccionar una de las varias funciones de *rehashing* que la herramienta ofrece: lineal, máximo primo, aleatoria y cuadrática. Todas las funciones de *hashing* implementadas han sido tomadas de [3, 5]. Nótese que en el caso de representación mediante tablas de *hash*, cambiando las funciones de *hashing* y/o *rehashing*, se obtienen diferentes representaciones. La herramienta también permite modificar el tamaño de las tablas de *hash*; esta acción se denomina redimensionar (y cada vez que se realiza se obtiene una nueva representación).

2.3 Información Estadística Generada por TADD

Cada una de las operaciones que se aplican durante un experimento, tiene un costo, en cuanto a número de comparaciones, intercambios y, en el caso de que la representación sea *Hashing* cerrado, número de colisiones. Estos costos son registrados y el usuario puede verlos y/o almacenarlos en memoria secundaria. Los costos pueden verse operación por operación y totalizados. Además, se calcula el número máximo y mínimo, promedio y desviación estándar de comparaciones, intercambios y colisiones ocurridos durante el experimento. Por otro lado, se mantiene un histórico de las estadísticas de los experimentos realizados en una sesión, las cuales pueden ser graficadas.

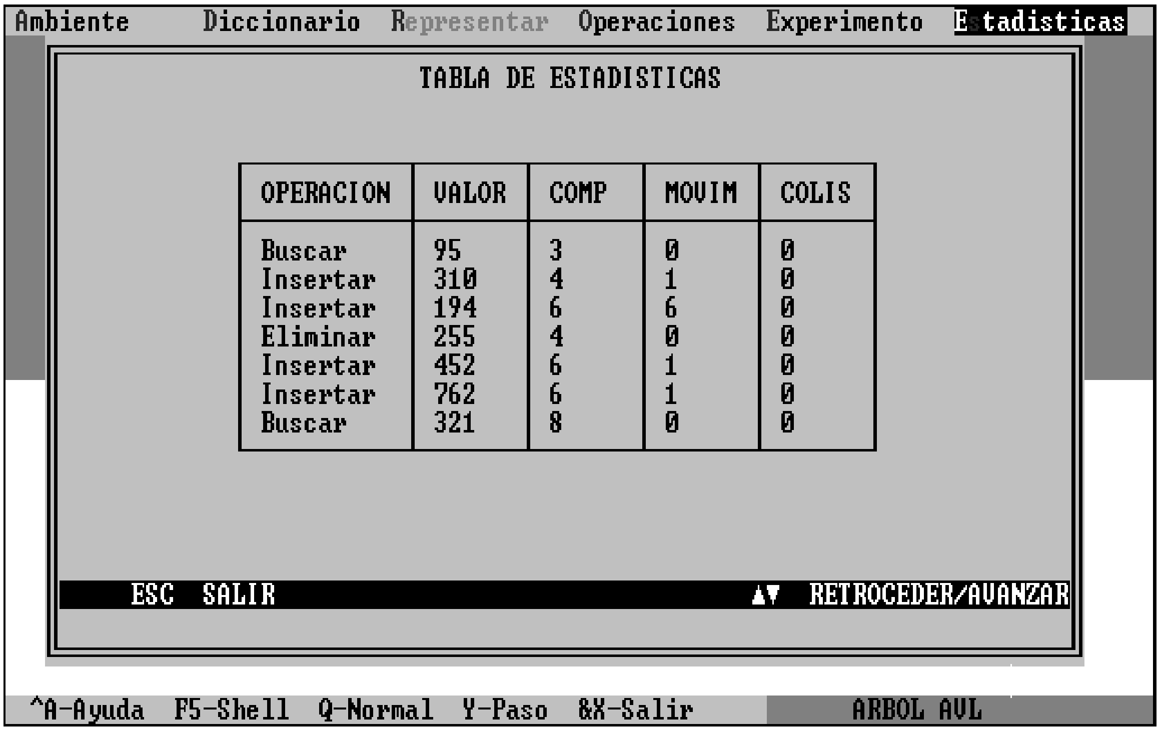
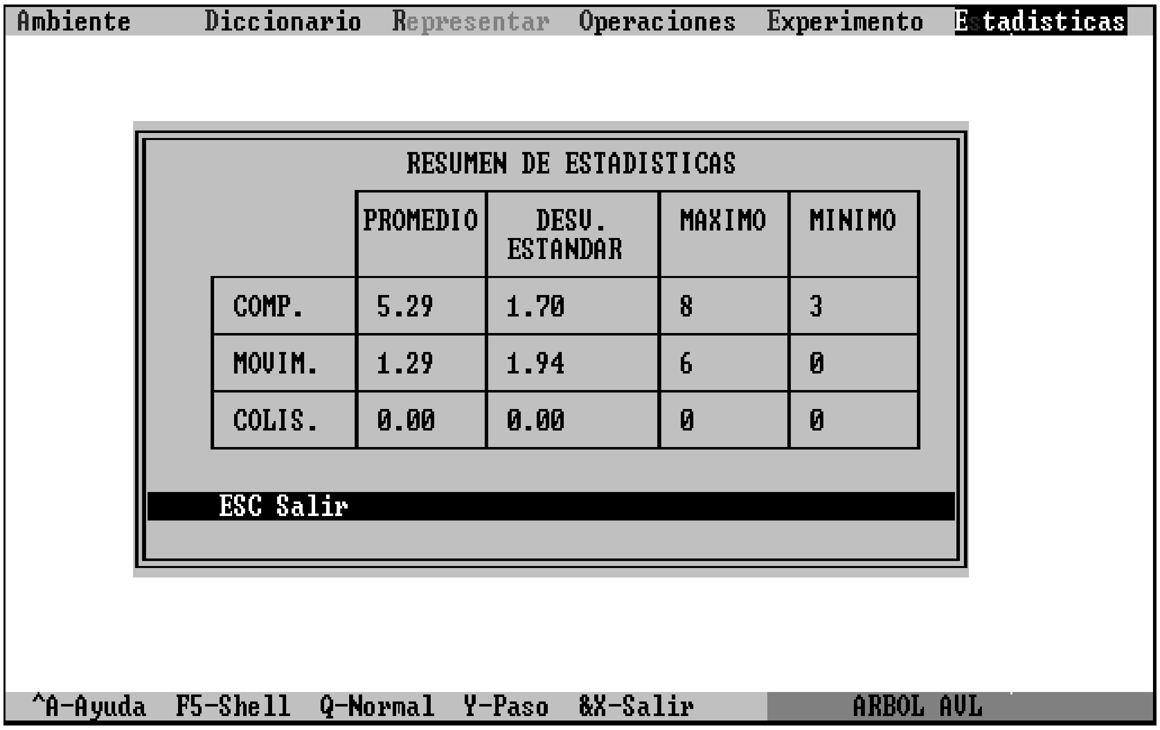


figura (1)
Tabla de Estadísticas Detalladas

En la fig. (1) se muestra una tabla de estadísticas generada por TADD. En ella puede verse el número de comparaciones, movimientos y colisiones para cada operación de un experimento realizado, así como los totales para cada uno de estos renglones.



	PROMEDIO	DESV. ESTANDAR	MAXIMO	MINIMO
COMP.	5.29	1.70	8	3
MOVIM.	1.29	1.94	6	0
COLIS.	0.00	0.00	0	0

figura (2)
Tabla de Estadísticas Generales

En la fig. (2) se muestra otro tipo de información: valor máximo y mínimo, promedio y desviación estándar de las comparaciones, movimientos y colisiones que se realizaron durante el experimento.

Esta información es de gran utilidad desde dos puntos de vista:

(i) Para realizar un análisis comparativo entre las distintas representaciones.

(ii) Para analizar el comportamiento de una misma representación en distintos escenarios.

Además, en el caso de las tablas de *Hash*, el usuario tiene la posibilidad de evaluar el comportamiento de las distintas funciones utilizadas, e incluso, puede evidenciar cómo influye el tamaño de la tabla que utilice en el desarrollo de sus experimentos.

2.4 Características Didácticas de TADD

En TADD es posible realizar los experimentos de tal manera que el usuario pueda visualizar cada uno de los pasos de las operaciones. Esta visualización se hace según el tipo de representación que se tiene en el momento de su realización. De esta manera, el usuario se familiariza tanto con el algoritmo de las operaciones, como con la estructura de datos que se utilizó para representar el diccionario.

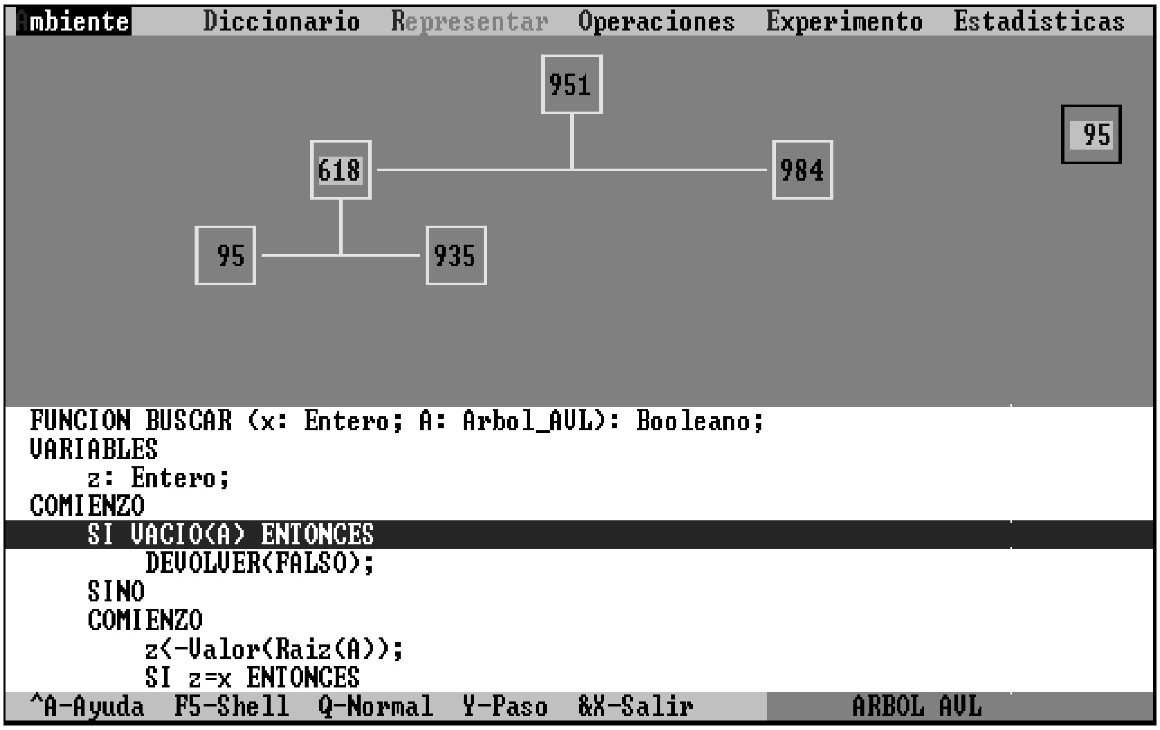


figura (3)
Un paso en la realización de un experimento
cuya representación es Arbol AVL.

La figura (3) muestra un paso en la realización de un experimento cuya representación es Árbol AVL. La operación que se está realizando es la búsqueda de la clave 95.

Además, TADD permite visualizar la ejecución de un experimento de diferentes modos: diccionario sólo, diccionario-algoritmo, o no visualización (en el ejemplo de la figura (3) el modo de visualización es diccionario-algoritmo). De esta manera, el usuario también tiene a su alcance los algoritmos y puede entender su funcionamiento. Para reforzar esto último, la herramienta también ofrece dos modos distintos para la realización de los experimentos: normal y paso a paso.

Con el objetivo de complementar su valor didáctico, la herramienta puede mostrar la especificación formal del Tipo Abstracto Diccionario y de los algoritmos asociados a las operaciones, de manera que el estudiante disponga tanto de las facilidades prácticas como de la información teórica que el tema requiere.

3 Conclusiones

La herramienta presentada permite realizar las operaciones básicas del Tipo Abstracto Diccionario para diferentes representaciones, registrando los costos de su ejecución. Además, ofrece al usuario la posibilidad de obtener estadísticas de dichos costos y generar gráficos que faciliten el análisis comparativo de los resultados obtenidos, lo que contribuye a la formación de criterios que sirven como base para escoger la representación más adecuada para el tipo diccionario, dado un problema que requiera de su uso.

Desde el punto de vista didáctico, a través de la animación de las operaciones, el usuario es capaz de entender claramente la manera en que éstas se realizan. Los distintos modos de visualización y ejecución que la herramienta ofrece, le permiten familiarizarse con los algoritmos de las operaciones.

TADD puede ser utilizada en los laboratorios de un curso básico de CS en el que se estudien tipos abstractos de datos, en particular, el tipo abstracto diccionario ya que contribuirá a que los estudiantes no sólo mejoren en la evaluación de las representaciones del tipo diccionario, sino a que, en general, comprendan mejor el concepto de un tipo abstracto de datos.

Una de las extensiones que se tienen previstas para TADD es la capacidad para generar automáticamente secuencias de operaciones que resulten de interés para los análisis que el estudiante debe realizar, como las que se presentan en [9]. Además se prevee continuar con el desarrollo de estas herramientas, basadas en el concepto de experimento, tanto para otros tipos abstractos de datos como para el estudio de estructuras de datos de almacenamiento en memoria secundaria (simulación).

Referencias

[1] Alfred Aho, John Hopcroft, Jeffrey Ullman. *Data Structures and Algorithms.* Addison Wesley Series in Computer Science. 1983.

[2] Philip Barker, Harry Yeates. *Introducing Computer Assisted Learning.* Prentice Hall International. 1985.

[3] A.T. Berztiss. *Data Structures. Theory and Practice.* 2^{nd} Ed. Academic Press. 1975.

[4] P. Borges. *HAAO: Una Herramienta para el Aprendizaje de Algoritmos de Ordenamiento.* Reporte Técnico No. CI-1992-010. Dpto. de Computación y Tecnología de la Información. Universidad Simón Bolívar. 1992

[5] G.H Gonnet, R.Baeza-Yates. *Handbook of Algorithms and Data Structures. In PASCAL and C.* 2^{nd} Ed. Addison Wesley.1991.

[6] L. Markezon, O. Vernet. *O uso de um Ambiente no Ensino de Grafos.* En actas del II Congreso Ibero-Americano de Educación Superior en Computación, Río de Janeiro, Brasil, 1992.

[7] A. Mayerhofer, C. Pereira. *Design of an Algorithm Simulation and Animation Enviroment.* SIGCSE Bulletin. Vol. 24, No. 2, pp. 7-14. June 1992.

[8] Oscar Meza, Maruja Ortega. *Grafos y Algoritmos.* Equinoccio, Ediciones de la Universidad Simón Bolívar. Caracas. 1993.

[9] Edelmira Pasarella, Sandra Zabala, Cristina Zoltan. *Un Enfoque Experimental para la Enseñanza de Tipos Abstractos de Datos.* Reporte Técnico No. CI-1995-002. Dpto de Computación y Tecnología de la Información. universidad Simón Bolívar, 1995

[10] Edelmira Pasarella, Cristina Zoltan. *Tipos Abstractos de Datos y Algoritmos.* Reporte Técnico No. CI-1994-003. Dpto. de Computación y Tecnología de la Información. Universidad Simón Bolívar, 1994.

[11] F. Procópio et al. *Especificacão de uma ferramenta para Verificação de erros lógicos em programas Pascal.* En actas del II Congreso Ibero-Americano de Educación Superior en Computación, Río de Janeiro, Brasil, 1992.

[12] R. Trahan, S. Rodger. *Simulation and Visualization Tools for Teaching Parallel Merge Sort.*, SIGCSE Bulletin. Twenty-Fourth SIGCSE Technical Symposium on Computer Science Education. Vol. 25, No. 1, pp. 237-241. March 1993.

[13] Nicklaus Wirth. *Algorithms + Data Structures = Programs.* Prentice Hall. 1976.

[14] Sandra Zabala, José M. Pérez, Alfredo Méndez. *Manual del Sistema TADD.* Reporte Técnico No. CI-1994-004. Dpto. de Computación y Tecnología de la Información. Universidad Simón Bolívar, 1994.