



# Proyecto final: Estructuras de datos

## Fundamentos de análisis y diseño de algoritmos

Carlos Andres Delgado S, Ing \*

Noviembre 2017

**Importante:** Recuerde entregar un informe en formato PDF y los códigos generados en un archivo comprimido en el campus virtual.

Así mismo, cuento con una herramienta de detección copia automática, por lo que, recomiendo hagan sus propias implementaciones a pesar de que las pueden encontrar en Internet. Recuerden explicar y analizar claramente cada punto, voy a rebajar si el informe tiene mala ortografía o mala redacción.

### 1. Matrices dispersas

Una matriz dispersa corresponde a un tipo de matriz de gran tamaño en la cual la cantidad de información relevante es baja en comparación con las dimensiones de la matriz. Típicamente, en este tipo de matrices la información no relevante se representa por el valor 0 (cero) y la información relevante por valores diferentes a 0. De esta manera, en este tipo de matrices es posible predecir que el porcentaje de ceros es alto. En general, considerar una matriz dispersa, osea, si la cantidad de ceros es lo suficientemente alta, depende del contexto específico en el cual se trabaje. Las matrices son estructuras de datos que en general son muy eficientes en cuanto al uso de memoria y de procesador. Sin embargo, la representación computacional de las matrices dispersas mediante el enfoque general tiene como consecuencia un uso ineficiente de la memoria y del tiempo de procesamiento. Esto se debe a que se utilizan muchas posiciones de memoria para almacenar ceros y se requiere procesar todas las posiciones de la matriz innecesariamente. En consecuencia, teniendo como objetivo mejorar la complejidad espacial y temporal de las operaciones y de la implementación específica de matrices dispersas, se han propuesto diversas representaciones. De esta manera, las matrices dispersas pueden ser definidas como una estructura de datos. El propósito de dichas representaciones es almacenar únicamente la información relevante de cada fila y columna de la matriz original con el propósito de utilizar menos memoria y agilizar las operaciones. En la siguiente sección se describen los principales enfoques de representación de matrices dispersas.

0	2	0	0	0	0	4
0	8	9	0	0	1	0
0	0	0	3	0	0	0
0	0	0	0	0	0	0
5	0	0	0	0	6	0
1	2	0	0	0	0	0
4	0	0	0	0	0	0
0	0	7	0	0	11	0

Cuadro 1: Ejemplo de matriz dispersa

### 2. Representaciones

En las siguientes secciones se describirán los enfoques de representación mas comunes para matrices dispersas. En adelante, se utilizara  $n$  y  $m$  para denotar el numero de las y columnas de la matriz completa, en tanto que se utilizara  $ne$  para denotar el numero de elementos diferentes de cero. Ademas, cada una de las representaciones sera ilustrada utilizando como referencia la matriz dispersa de la tabla 1

---

\*carlos.andres.delgado@correounivalle.edu.co

## 2.1. Formato Coordinado

La representación de matrices dispersas mediante el formato coordinado almacena los datos distintos de cero de la matriz dispersa junto con su ubicación en la matriz, i.e., el índice de su fila y su columna. Para esto se utilizan 3 vectores de tamaño  $ne$ . En el primer vector se almacenan los valores distintos de cero de la matriz original mientras que en el segundo y tercer vector se almacenan los índices de la fila y la columna para cada uno de los valores distintos de cero.

Los siguientes serían los valores en los vectores para la representación en formato coordinado de la matriz de ejemplo son:

```
valores = [2 4 8 9 1 3 5 6 1 2 4 7 11]
filas = [0 0 1 1 1 2 4 4 5 5 6 7 7]
columnas = [1 6 1 2 5 3 0 5 0 1 0 2 5]
```

## 2.2. Formato Comprimido

### 2.2.1. Formato comprimido por fila (CSR)

El formato comprimido por fila (**CSR**) representa una matriz dispersa por medio de 3 vectores. El primer vector tiene tamaño  $ne$  y almacena los valores distintos de cero de la matriz original organizados fila por fila. En el segundo vector, que también tiene tamaño  $ne$ , se almacenan los índices de las columnas en las que están cada uno de los valores del primer vector en la matriz original. En tanto que, el tercer vector tiene tamaño  $n + 1$  y almacena la posición donde empiezan los valores de cada fila en el segundo vector (vector de columnas). Para el caso del ejemplo se tiene:

```
valores = [2 4 8 9 1 3 5 6 1 2 4 7 11]
cols = [1 6 1 2 5 3 0 5 0 1 0 2 5]
cfilas = [0 0 1 1 1 2 4 4 5 5 6 7 7]
```

Observe que el tercer vector (vector **cfilas**) se puede interpretar asociando el valor  $c_i$  y el valor  $c_{i+1} - 1$  como el índice inicial y el índice final en el vector de columnas asociados a la fila  $i$ . Es importante resaltar que en esta representación cuando una fila no contiene ningún valor distinto a cero entonces el valor asociado a dicha fila en el tercer vector (vector **cfilas**) es el mismo de la fila anterior. Además, el último valor en el tercer vector corresponde al último índice válido en el segundo vector más 1.

### 2.2.2. Formato comprimido por columna (CSC)

El formato comprimido por columna (**CSC**) también representa una matriz dispersa por medio de 3 vectores. Este formato es análogo al formato CSR y utiliza también 3 vectores. El primer vector tiene tamaño  $ne$  y almacena los valores no nulos en la matriz original organizados por columna. El segundo vector tiene tamaño  $ne$  y almacén los índices de las filas en las que están cada uno de los valores del primer vector. Mientras que el tercer vector tiene tamaño  $m+1$  y almacena la posición donde empiezan los valores de cada columna en el segundo vector (vector de filas). Para el caso del ejemplo la representación es:

```
valores = [5 1 4 2 8 2 9 7 3 1 6 11 4]
filas = [4 5 6 0 1 5 1 7 2 1 4 7 0]
ccolumnas = [0 0 0 1 1 1 2 2 3 5 5 5 6]
```

Observe que el tercer vector (vector **ccolumnas**) se puede interpretar asociando el valor  $c_i$  y el valor  $c_{i+1} - 1$  como el índice inicial y el índice final en el vector de filas asociados a la columna  $i$ .

Como en la representación CSR, cuando una columna no contiene ningún valor distinto a cero entonces el valor asociado a dicha columna en el tercer vector (vector **ccolumnas**) es el mismo de la columna anterior.

## 3. Operaciones

Las representaciones de matrices dispersas deben permitir:

1. **Crear una matriz completa.** Pasar de la representación a la forma matricial
2. **Obtener representación.** Pasar de la forma matricial a la representación
3. **Obtener elemento:** Dada una representación y una posición  $i,j$  debe retornarse el valor asociado.
4. **Obtener fila:** Dada una representación y una fila  $i$ , debe retornarse la fila asociada

5. **Obtener columna:** Dada una presentación y una columna  $j$ , debe retornarse la columna asociada.
6. **Modificar posición:** Dada una representación, un posición  $i,j$  y un elemento, debe modificarse la representación para ingresar este elemento.
7. **Suma de matrices:** Dadas dos representaciones (del mismo tipo) permitir la suma de las dos matrices (Ambas deben ser del mismo tamaño)
8. **Matriz transpuesta:** Dada una representación, retornar la representación de la matriz transpuesta

## 4. Proyecto

Este proyecto puede ser realizado en grupos de hasta 3 personas. El proyecto debe ser sustentado, la cual tiene una nota entre 0 y 1 individual, la cual será multiplicada por la nota del proyecto.

1. **(30 %)** Implementación de las funciones de creación de las representaciones. Estas reciben una matriz y retorna la representación
  - a) Función para generar representación en formato coordenadas
  - b) Función para generar representación en formato CSC
  - c) Función para generar representación en formato CSR
2. **(15 %)** Implementación de las operaciones 2,3,4,5 y 6
3. **(15 %)** Implementación de la suma de matrices. Esta función recibe una representación y retorna la representación de la suma Debe funcionar para las 3 implementacion
4. **(15 %)** Implementación de la matriz transpuesta. Esta función recibe una representación y retorna la representación de la transpuesta. Debe funcioanr para las implementaciones
5. **(25 %)** Realizar una comparación entre las tres implementaciones, teniendo en cuenta la complejidad de las operaciones y aspectos generales de ellas. Realice las conclusiones del ejercicios en base a este análisis. Recuerde que las conclusiones son producto del ejercicio realizado.

Como parte de la nota, debe realizar un informe **en formato PDF** que cumpla:

1. El informe debe tener los nombres y códigos de los integrantes del grupo
2. Explique el proceso que realiza para generar la representación dada una matriz cualquiera. Realice un ejemplo adicional al mostrado en este enunciado de las 3 representaciones. Esto es parte de la nota del item 1. Explique los aspectos más relevantes de su implementación, tales como estrategias y estructuras de datos utilizadas. **Importante:** Evite copiar código, a menos que sea estrictamente necesario.
3. Explicar cada una de las funciones implementadas. Explicar las estrategias utilizadas y estructuras de datos. **Importante:** Evite copiar código, a menos que sea estrictamente necesario. Parte de los items 2,3 y 4.
4. Análisis comparativo, conclusiones y bibliografía. Item 5.