

## Tarea Número 1. Bases de Datos avanzadas.

En esta tarea usted debe 1. implementar un map-reduce en erlang y 2. utilizar esta implementación de map-reduce para implementar el algoritmo de pagerank de forma eficiente.

### Map-Reduce simple.

El objetivo de esta parte de la tarea es lograr la implementación de un map-reduce de un vector por una matriz, puede suponer que la matriz se encuentra en un archivo como una secuencia de tuplas, un ejemplo de archivo puede ser.

```
6.  
{3,6}  
{2,5}  
{1}  
{}  
{5,6}  
{2,3,4}
```

Este archivo se refiere a una matriz de 6x6 de la forma:

	1	2	3	4	5	6
1	0	0	1	0	0	0
2	0	0.5	0	0	0	0.33
3	0.5	0	0	0	0	0.33
4	0	0	0	0	0	0.33
5	0	0.5	0	0	0.5	0
6	0.5	0	0	0	0.5	0

El vector por el cual debe multiplicar esta matriz puede suponer que se encuentra en otro archivo como una secuencia de números. Su programa debe poder ejecutarse de la siguiente manera:

> multMatriz(Nodos, Trabajadores, Reductores, Matrix, Vector).

Donde:

1. Nodos es la lista de nombres de los nodos donde va a ejecutar los trabajadores.
2. Trabajadores es la cantidad total de trabajadores que debe distribuirse equitativamente entre los nodos.
3. Reductores es la cantidad total de reduce tasks, igual debe distribuirla equitativamente entre los nodos.
4. Matrix es el nombre del archivo donde se encuentra la matriz.
5. Vector es el nombre del archivo donde se encuentra el vector a multiplicar.

### Map-Reduce modificado.

Implemente el map-reduce utilizando un factor beta para garantizar la ergodicidad de la matriz de conexiones.

### Page-Rank.

Una vez logrado el map-reduce debe implementar un page-rank que itere sobre el map-reduce para lograr el valor del vector final.

Fecha de entrega 8 de Octubre