

# Creación de árboles

## 1. Definición del problema:

Se debe realizar una búsqueda de forma rápida entre la lista de estudiantes del poli, dada su cédula retornar el nombre de la persona.

## 2. Propuesta de solución con texto y gráficas:

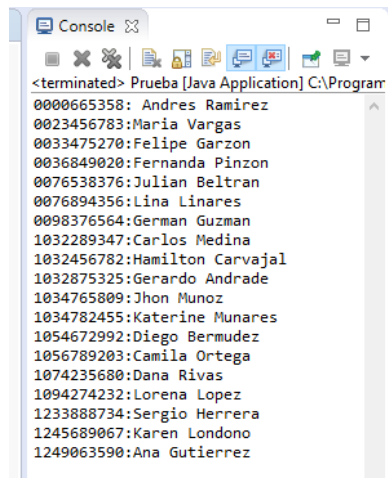
Teniendo un archivo plano que contiene los nombres y las cédulas de los estudiantes del poli, se cargan los datos del archivo en un arreglo llamado estudiante mediante un objeto de tipo estudiante, a partir del arreglo se creara un árbol binario.

Este árbol permitirá insertar nuevos nodos, sobre esto se realizara la búsqueda del estudiante a partir de su cédula.

### - Base de datos de los estudiantes en archivo plano

1		
2	<b>Cédula</b>	<b>Nombre</b>
3	1249063590	Ana Gutierrez
4	665358	ándres Ramirez
5	1032289347	Carlos Medina
6	1056789203	Camila Ortega
7	1054672992	Diego Bermudez
8	1074235680	Dana Rivas
9	33475270	Felipe Garzón
10	36849020	Fernanda Pinzón
11	1032875325	Gerardo Andrade
12	98376564	German Guzmán
13	1032456782	Hamilton Carvajal
14	76538376	Julián Beltrán
15	1034765809	Jhon Muñoz
16	1245689067	Karen Londoño
17	1034782455	Katerine Munares
18	76894356	Lina Linares
19	1094274232	Lorena López
20	23456783	María Vargas
21	1233888734	Sergio Herrera
22		

### - Base de datos de los estudiantes en consola.



```
<terminated> Prueba [Java Application] C:\Program
0000665358: Andres Ramirez
0023456783: Maria Vargas
0033475270: Felipe Garzon
0036849020: Fernanda Pinzon
0076538376: Julian Beltran
0076894356: Lina Linares
0098376564: German Guzman
1032289347: Carlos Medina
1032456782: Hamilton Carvajal
1032875325: Gerardo Andrade
1034765809: Jhon Munoz
1034782455: Katerine Munares
1054672992: Diego Bermudez
1056789203: Camila Ortega
1074235680: Dana Rivas
1094274232: Lorena Lopez
1233888734: Sergio Herrera
1245689067: Karen Londono
1249063590: Ana Gutierrez
```

3. Solución propuesta en código fuente, debe colgar su código fuente en GIST de github.com.  
<https://github.com/NathaVargas/ArbolBinario>

4. Conclusiones y soluciones alternativas:

- Conclusiones

De este ejercicio se puede concluir que el programar de una manera estructurada nos ayuda a ser más eficientes, además el código fuente es más fácil de entender para las demás personas, de más fácil mantenimiento y abierto a mejoras de una forma sencilla.

- Soluciones alternativas

Este ejercicio se puede solucionar simplemente con un arreglo o arraylist en el cual se cargan los datos del archivo plano, luego se realiza la búsqueda recorriendo el arreglo y comparando los datos.

5. Bibliografía:

<http://moodlep16a.mipoli.co/course/view.php?id=155>  
<http://www.javaya.com.ar/detalleconcepto.php?codigo=127&inicio=40>  
<http://cristiantorresalfaro.blogspot.com.co/2012/06/buscar-datos-en-arreglo-metodo-busqueda.html>  
<http://usandojava.blogspot.com.co/2012/11/busqueda-binaria-en-un-arreglo-usando.html>  
[https://www.google.com.co/search?q=arreglo+de+objetos+java&oq=arreglo+de+obj&aqs=chrome.3.0j69i65j69i57j0l2j69i61.7243j0j7&sourceid=chrome&ie=UTF-8#q=como+partir+una+cadena+en+java+String.split\(...\)](https://www.google.com.co/search?q=arreglo+de+objetos+java&oq=arreglo+de+obj&aqs=chrome.3.0j69i65j69i57j0l2j69i61.7243j0j7&sourceid=chrome&ie=UTF-8#q=como+partir+una+cadena+en+java+String.split(...))  
<http://es.wikihow.com/comparar-cadenas-de-caracteres-en-Java>