

# INSTITUTO TECNOLÓGICO DE COSTA RICA

PROYECTO DE INVESTIGACIÓN DE OPERACIONES

PROYECTO 2

---

## Árboles Binarios

---

### Estudiantes

Jason Barrantes Arce  
2015048456

Steven Bonilla Zúñiga  
2015056296

### Profesor

FRANCISCO TORRES ROJAS

### Modo Ejemplo:

Se resolverá un problema general por medio de diversos algoritmos que nos permitan encontrar una solución a ese problema. El problema que se nos plantea es sobre árboles de búsqueda binarios.

Hay que ordenar una serie de valores o llaves en forma de árbol binario, de manera que el nivel de búsqueda promedio sea el más óptimo. Tenemos una serie de llaves que quieren acomodarse en una estructura de árbol, pero esas llaves deben tener un respectivo peso y un carácter único que será asignado de manera aleatoria.

Restricciones:

- **Estructura:** Se formará un árbol binario óptimo.
- **Llaves:** Se generarán 6 llaves de forma aleatoria.
- **Peso:** Varía entre  $1 \leq C_i \leq 1000$ .
- Los caracteres ASCII varían.

Los dos algoritmos que vamos a implementar son:

- **Algoritmo de Búsqueda Dinámica:** Algoritmo para el ABB óptimo.
- **Algoritmo Greedy Básico:** Cada vez se escoge la llave de máxima probabilidad para que sea la raíz del árbol.

En el caso de programación dinámica ya que nuestro objetivo es minimizar el costo promedio de la búsqueda, usamos la fórmula:

$$\text{MIN}(Z) = \sum_{i=1}^n c_i p_i$$

Que está sujeto a:

$$\sum p_i \equiv 1$$

Con cada  $c_i = 1, 2, \dots, n$

Se muestra a continuación la tabla de objetos con su respectivo costo (peso) y probabilidad que fueron asignados aleatoriamente cumpliendo con las restricciones:

Letra	Peso	Probabilidad
Object \	591	0.18
Object +	583	0.18
Object L	849	0.26
Object k	244	0.07
Object p	842	0.26
Object B	168	0.05

## 1. Algoritmo AAB Óptimo

Es un problema de solución de búsqueda óptima: En nuestro caso, queremos minimizar el costo de la búsqueda promedio. Para solucionar el problema vamos a hacer uso de una tabla  $(n+1) \times (n+1)$ , donde  $n$  es la cantidad de objetos o llaves disponibles. Como se menciona en las restricciones del problema  $n = 6$  por lo que tendremos dos tablas (7x7). La tabla A donde estará el costo promedio y la tabla R donde estará los índices de búsqueda más rápidos.

### Fórmula Matemática

$$\text{MIN}(Z) = \sum_{i=1}^6 c_i p_i$$

Sujeto a:

$$\sum p_i \leq 1$$

En otras palabras tendremos

$$(p_1 \approx 0,1803) + (p_2 \approx 0,1779) + (p_3 \approx 0,2591) + (p_4 \approx 0,0745) + (p_5 \approx 0,2569) + (p_6 \approx 0,0513) = 1$$

Ahora proseguimos realizando la tabla dinámica.

**Tabla A:**

X	0	1	2	3	4	5	6
1	0.00	0.18	0.54	1.06	1.28	1.89	2.04
2	–	0.00	0.18	0.61	0.76	1.35	1.51
3	–	–	0.00	0.26	0.41	1.00	1.10
4	–	–	–	0.00	0.07	0.41	0.51
5	–	–	–	–	0.00	0.26	0.36
6	–	–	–	–	–	0.00	0.05
7	–	–	–	–	–	–	0.00

**Tabla R:**

X	0	1	2	3	4	5	6
1	0	1	1	2	2	3	3
2	-	0	2	3	3	3	3
3	-	-	0	3	3	3	5
4	-	-	-	0	4	5	5
5	-	-	-	-	0	5	5
6	-	-	-	-	-	0	6
7	-	-	-	-	-	-	0

El algoritmo tarda aproximadamente: 0.005000 segundos en ejecutarse