

COLISIONES DE ABEJAS

*Esteban Zapata
Camilo Restrepo
Medellín, Mayo 15*

Ordenamiento QuickSort respect a eje x:

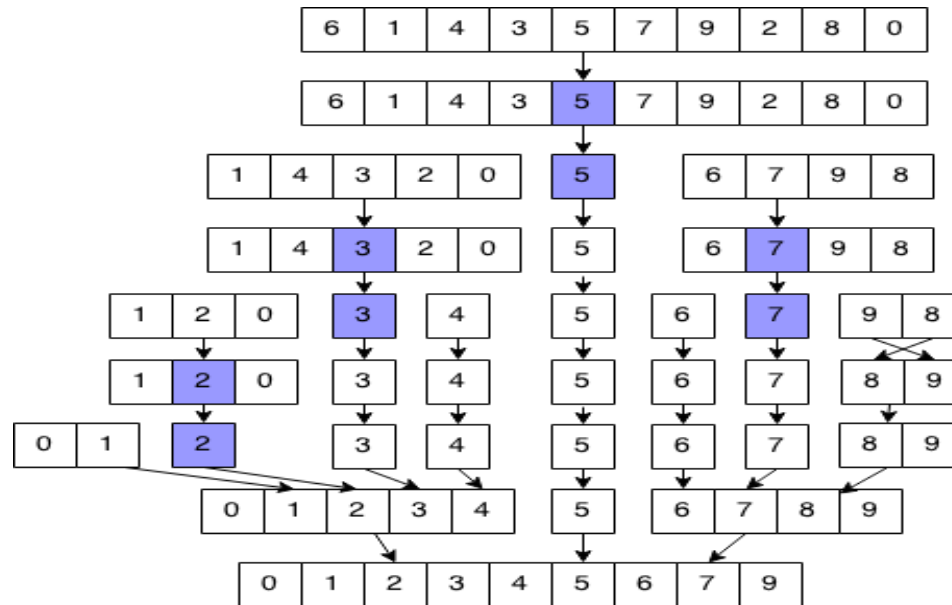


Grafico 1: Representa un ejemplo de un ordenamiento QuickSort

Estructura de datos diseñada:

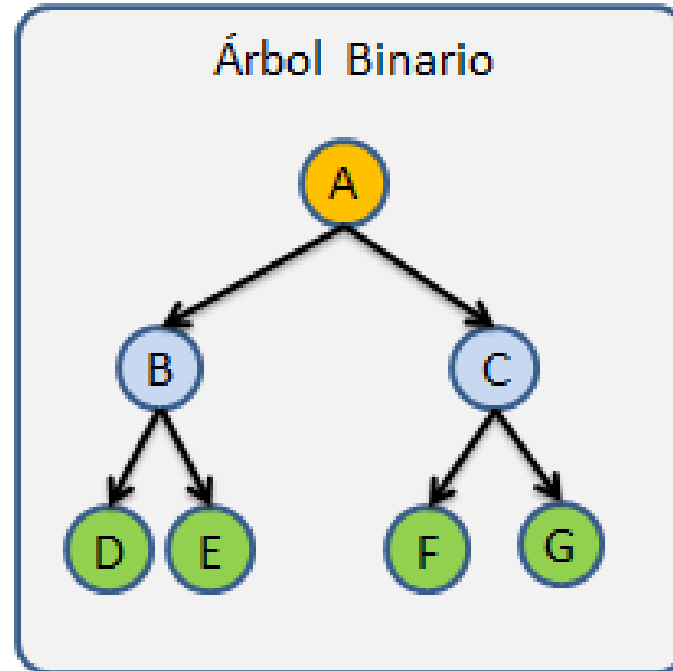


Grafico 2: Representa la division del ArrayList de Abejas

Complejidad

Clases	Complejidad
LeerTxt()	$O(n)$
Ordenador()	$O(n \log(n))$
Tree()	$O(n \log(n))$

Tabla 1: Complejidad de las clases

Criteria de diseño

- El proceso de ordenamiento tiene una complejidad de $O(n\log(n))$
- En la solución las operaciones básicas se requiere tener como complejidad $O(1)$
- En la división de ArrayList la complejidad no debe ser mayor a $O(n)$
- El análisis de colisiones se subdivide en dos análisis con complejidad $O(n\log(n))$

Consumo de tiempo y memoria

Operaciones	10 Bees (s)	100 Bees (s)	1.000 Bees (s)	10.000 Bees (s)	100.000 Bees (s)
Leer	0.004	0.043	0.48	5.31	56.85
Ordenar	0.006	0.061	0,68	7.25	76,41
Colisiones	0.008	0.082	0,96	11.85	113,87

Tabla 2: Se muestra cuanto tiempo tarda en realizar las operaciones

Funcionamiento

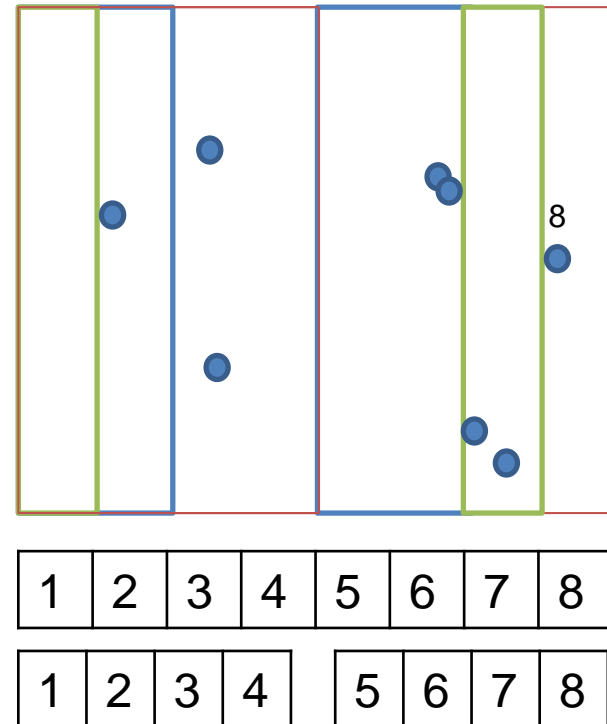
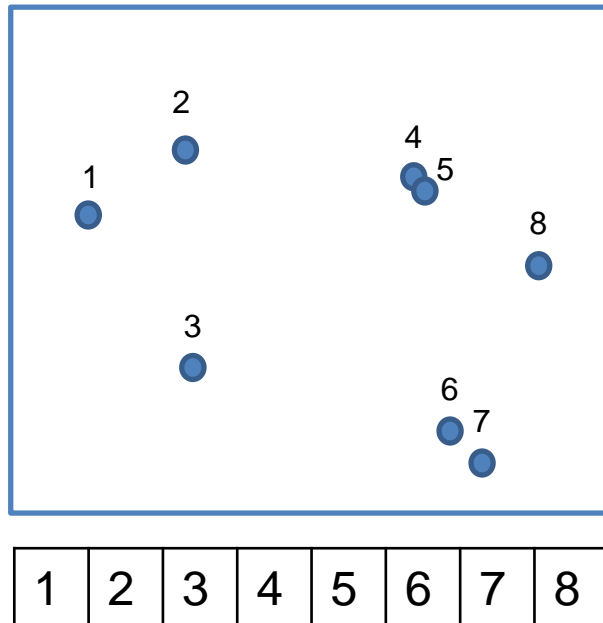


Grafico 3: Representa la division del ArrayList de Abejas