

Algoritmo	Complexidade	Lista Pequena (100 Elementos)
Binary Search	$O(\log n)$	Rápido
Interpolation Search	$O(\log \log n)$	Lento em distribuição Irregular
Jump Search	$O(\text{RaizQuadrada } N)$	Mediano
Exponential Search	$O(\log i) + O(\log n)$	Rápido

Algoritmo	Tempo	Comparações
Selection Sort	0.000005 Segundos	21
Quick Sort	0.0000010 Segundos	13
Bucket Sort	0.000007 Segundos	7
Radix Sort	0.000012 Segundos	35

Análise de Complexidade	Tipo	Melhor Caso
Binary Search	Busca	$O(1)$
Interpolation Search	Busca	$O(1)$
Jump Search	Busca	$O(1)$
Exponential Search	Busca	$O(1)$
Merge Sort	Ordenação	$O(n \log n)$
Selection Sort	Ordenação	$O(n^2)$
Bucket Sort	Ordenação	$O(n+k)$
Radix Sort	Ordenação	$O(nk)$
Quick Sort	Ordenação	$O(n \log n)$
Ternary Search	Ordenação	$O(1)$

Lista Média (10.000 Elementos)	Lista Grande (1.000.000 Elementos)
Muito rápido	Muito rápido
Rápido com distribuição uniforme	Muito rápido com dados uniformes
Lento	Muito lento
Rápido	Muito rápido

Lista
[1,3,5,7,9,11,13]
[1,3,5,7,9,11,13]
[1,3,5,7,9,11,13]
[1,3,5,7,9,11,13]

Médio Caso	Pior Caso
$O(\log n)$	$O(\log n)$
$O(\log \log n)$	$O(n)$
$O(\sqrt{n})$	$O(\sqrt{n})$
$O(\log n)$	$O(\log n)$
$O(n \log n)$	$O(n \log n)$
$O(n^2)$	$O(n^2)$
$O(n^2)$	$O(n^2)$
$O(nk)$	$O(nk)$
$O(n \log n)$	$O(n^2)$
$O(\log_3 n)$	$O(\log_3 n)$