|  |  |
| --- | --- |
| **Nombre del algoritmo** | ShellSort |
| **Mejor caso** | Cuando el arreglo está ordenado |
| **Complejidad mejor caso** | O(Nlog(N)) |
| **Peor caso** | Cuando está muy desordenado |
| **Complejidad peor caso** | O(N^2) |
| **Inplace** | Si |
| **Adaptativo** | Si |
| **Estable** | No |

|  |  |
| --- | --- |
| **Nombre del algoritmo** | MergeSort |
| **Mejor caso** | Cuando el arreglo ya está ordenado |
| **Complejidad mejor caso** | O(N) |
| **Peor caso** | Cuando el arreglo está totalmente desordenado |
| **Complejidad peor caso** | O(Nlog(N)) |
| **Inplace** | No |
| **Adaptativo** | Si |
| **Estable** | Si |

|  |  |
| --- | --- |
| **Nombre del algoritmo** | QuickSort |
| **Mejor caso** | El pivote termina en el centro de la lista |
| **Complejidad mejor caso** | O(Nlog(N)) |
| **Peor caso** | Cuando el arreglo está casi ordenado y el pivote acaba en un extremo |
| **Complejidad peor caso** | O(N^2) |
| **Inplace** | Si |
| **Adaptativo** | No |
| **Estable** | No |

|  |  |  |  |
| --- | --- | --- | --- |
|  | ShellSort | MergeSort | QuickSort |
| Tiempo 1 | 542 | 130 | 430 |
| Tiempo 2 | 457 | 128 | 373 |
| Tiempo 3 | 450 | 128 | 413 |
| Promedio | 483.00 | 128.67 | 405.33 |

Utilizando la carga de comparendos y haciendo las pruebas unitarias el algoritmo más efectivo es merge sort, luego quicksort y por ultimo Shell sort.