|  |  |  |  |
| --- | --- | --- | --- |
|  | Bubble | Insertion | Merge |
| Partial10 | 0 | 0 | 0 |
| Partial1000 | 0.009001 | 0.004 | 0 |
| Partial10000 | 0.848048 | 0.387022 | 0.001 |
| Partial100000 | 84.9249 | 38.6222 | 0.010005 |
|  |  |  |  |

|  |  |  |  |
| --- | --- | --- | --- |
|  | Bubble | Insertion | Merge |
| Unique10 | 0 | 0 | 0 |
| Unique1000 | 0.007 | 0.002 | 0.001 |
| Unique10000 | 0.762044 | 0.198011 | 0.001 |
| Unique100000 | 76.5304 | 19.8001 | 0.014001 |

|  |  |  |  |
| --- | --- | --- | --- |
|  | Bubble | Insertion | Merge |
| Reversed10 | 0 | 0 | 0 |
| Reversed1000 | 0.009001 | 0.004 | 0 |
| Reversed10000 | 0.854049 | 0.397023 | 0.001 |
| Reversed100000 | 0.001 | 0.001 | 0.008 |

|  |  |  |  |
| --- | --- | --- | --- |
|  | Bubble | Insertion | Merge |
| Random10 | 0 | 0 | 0 |
| Random1000 | 0.008005 | 0.002001 | 0 |
| Random10000 | 0.761043 | 0.197011 | 0.001 |
| Random100000 | 76.5814 | 19.7721 | 0.014001 |

Overall merge sort was the fastest sorting algorithm, followed by insertion sort, then bubble sort. All the algorithms were slower on the partially sorted set and the reversed set.