|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Radix Sort** | **Quick Sort** | | **algorithm** | |
| **Time(ms)** | | **#comparison** |
| 531 | 125 | 27061 | input1.txt | **inputs** |
| 1290 | 205 | 62928 | input2.txt |
| 1831 | 279 | 146556 | input3.txt |
| 2555 | 329 | 195929 | input4.txt |
| 3236 | 520 | 292926 | input5.txt |
| 3950 | 466 | 370222 | input6.txt |
| 4335 | 533 | 411463 | input7.txt |
| 4180 | 566 | 432422 | input8.txt |
| 3739 | 617 | 604914 | input9.txt |
| 3233 | 869 | 577686 | input10.txt |
| 2958 | 911 | 673427 | input11.txt |

اگر k کم باشد و n زیاد باشد radix از quick سریع تر است چون quick به تعداد n وابستگی بیشتری دارد و radix نیز به طول رشته ماکسیمم وابستگی بیشتری دارد.

در سوال ۴ نمیتوان از merge sort استفاده کرد زیرا ناپایدار است. و ما برای استفاده از radix sort به یک مرتب سازی پایدار نیازمندیم.