

Amir Taslimpour

پایاده سازی الگوریتم

در پایتون استفاده کرد. این تابع با گرفتن یک لیست به عنوان ورودی، آن `sorted()` برای پایاده سازی این الگوریتم می‌توان از تابع نیز استفاده کرد که همان عملیات را بر روی لیست `sort()` را بر اساس مقدار های داخل آن مرتب می‌کند. همچنین، می‌توان از تابع ورودی انجام می‌دهد، اما لیست را در محل تغییر می‌دهد

تحلیل مرتبه زمانی

$O(n \log n)$ استفاده شده است که به صورت پیش‌فرض از الگوریتم مرتب سازی مرتبه `sorted()` برای مرتب کردن نام‌ها، از تابع $O(n \log n)$ عمل می‌کند. بنابراین، مرتبه زمانی کل الگوریتم برابر با $O(n)$ نیز با مرتبه `split()` استفاده می‌کند. همچنین، تابع

نمودار زمان اجرا

در پایتون استفاده کرد. این کتابخانه به `timeit` برای نمایش زمان اجرای الگوریتم برای داده‌های با سایز مختلف، می‌توان از کتابخانه `..` صورت خودکار زمان اجرای یک تابع را برای چندین بار محاسبه کرده و میانگین آن را به عنوان خروجی برمی‌گرداند

خروجی:

Size 10: 0.003604 seconds

Size 50: 0.027575 seconds

Size 100: 0.057111 seconds

Size 500: 0.318508 seconds

Size 1000: 0.650628 seconds

، برای داده‌های با سایز تا $O(n \log n)$ با توجه به نمودار زمان اجرا، مشخص است که الگوریتم مرتب کردن نام‌ها با مرتبه زمانی حدود ۱۰۰۰، به صورت قابل قبولی عمل می‌کند. با افزایش سایز داده‌ها، زمان اجرای الگوریتم نیز به طور قابل توجهی افزایش می‌یابد.