

کلاس $\text{Map}\langle K, V \rangle$

متد $\text{put}(K \text{ key}, V \text{ value})$

- پیچیدگی زمانی:
- o پیدا کردن شاخص $O(1)$
- o جستجوی کلید $O(L)$
- o درج عنصر $O(1)$
- o پیچیدگی کلی $O(n)$
- پیچیدگی فضایی
- o فضای اضافی برای ذخیره کلید و مقدار $O(1)$

متد $\text{get}(K \text{ key})$

- پیچیدگی زمانی:
- o پیدا کردن شاخص $O(1)$
- o جستجو $O(L)$
- o پیچیدگی کلی $O(n)$
- پیچیدگی فضایی:
- o بدون نیاز به فضای اضافی

متد $\text{resize}()$

- پیچیدگی زمانی
- o بازسازی تمام عناصر $O(n)$
- پیچیدگی فضایی
- o ایجاد آرایه جدید دو برابر بزرگ تر $O(2n)=O(n)$

متدهای دیگر

- remove , keySet , values هر کدام پیچیدگی $O(n)$ دارند

کلاس Preprocessor

متد preprocessDocuments(String folderPath)

- پیچیدگی زمانی
 - o خواندن تمام فایل ها $O(f)$
 - o پردازش محتوای هر فایل $O(n)$
 - o پیچیدگی کلی $O(f+n)$
- پیچیدگی فضایی
 - o ذخیره محتوای اصلی و پردازش شده $O(n)$

متد cleanContent(String content)

- پیچیدگی زمانی
 - o تبدیل به حروف کوچک $O(n)$
 - o حذف کاراکترهای غیرمجاز $O(n)$
 - o شکستن به کلمات $O(n)$
 - o پیچیدگی کلی $O(n)$
- پیچیدگی فضایی
 - o ایجاد لیستی از کلمات $O(n)$

کلاس SearchEngine

متد preprocessDocuments(String folderPath)

- پیچیدگی زمانی
 - o ساخت ایندکس معکوس $O(n)$
 - o پیچیدگی کلی $O(f+n)$

- پیچیدگی فضایی
- o ذخیره ایندکس معکوس $O(n)$
- متد `search(String query)`
- پیچیدگی زمانی
- o تقسیم پرس وجو به کلمات $O(q)$
- o جستجوی هر کلمه در ایندکس $O(q \cdot l)$
- o پیچیدگی کلی $O(q \cdot l)$
- پیچیدگی فضایی
- o ذخیره مجموعه های نتایج میانی $O(n)$
- متد `validateQuery(String query)`
- پیچیدگی زمانی
- o تقسیم کلمات و بررسی تکرار عملگرها $O(q)$:
- o پیچیدگی کلی $O(q)$
- پیچیدگی فضایی
- o ذخیره کلمات و عملگرها در $O(q)$ Map

کلاس `InvertedIndex`

متد `buildIndex(Map<Integer, List<String>> documents)`

- پیچیدگی زمانی
- o پیمایش تمام کلمات در تمام فایل ها $O(n)$
- o افزودن هر کلمه به ایندکس $O(1)$
- o پیچیدگی کلی $O(n)$
- پیچیدگی فضایی

o ذخیره ایندکس معکوس $O(n)$

متد `getDocuments(String word)`

• پیچیدگی زمانی

o جستجوی کلمه در ایندکس $O(1)$

• پیچیدگی فضایی

o بدون نیاز به فضای اضافی