

بسمه تعالی

آرین خلوصی - ۹۵۲۴۲۸۳

پروژه دوم درس طراحی الگوریتم استاد احمدی

پروژه ی پیاده سازی شده دومین سوال پروژه است (مساله جیمز باند)

در برنامه نویسی پویا برای اینکه از محاسبه مجدد نتایج بازگشتی جلوگیری کنیم نتایج را مثلا در یک آرایه یا ساختمان داده دیگری ذخیره میکنیم. در اینجا ابتدا یک آرایه $n \times n$ که n طول رشته است، می سازیم و در کد از آن به عنوان dp table یا به اختصار dp استفاده می کنیم. هر عنصر این آرایه می تواند شامل مقادیر ۱- تا ۱ باشد.

$dp[i][j]$ بیانگر این است که زیررشته $s[i][j]$ در دیکشنری موجود است یا خیر.

۱- یعنی عنصر موردنظر هنوز دیده نشده. صفر به معنای بی معنی بودن کلمه و ۱ به معنای معنادار بودن آن است. بطور پیشفرض همه ی المنت های آرایه ۱- هستند.

طرز کار الگوریتم به این صورت است که رشته را ابتدا چک میکنیم که در دیکشنری باشد یا خیر. اگر بود که در جدولمان آن را با ۱ علامت میزنیم در غیراینصورت یک حلقه از عنصر ابتدایی رشته تا انتهای رشته میزنیم. اگر زیررشته مان متناظرا در جدولمان علامت گذاری شده بود که مقدارش را برمیگردانیم در غیر اینصورت بصورت بازگشتی مقدارش را محاسبه میکنیم. به این صورت که رشته را تکه تکه میکنیم تا به اجزای معنادار برسیم. در هر پیمایش اگر $s[start, i]$, $s[i+1, end]$ هر دو در دیکشنری موجود بودند مقدار متناظر آن را در جدول علامت میزنیم. (برای توضیح دقیق تر هر بخش در کد سعی کرده ام کامنت مناسب بگذارم).

پیچیدگی فضایی این الگوریتم به خاطر سایز آرایه n^2 است.

تعداد زیررشته ها n^2 است. عملیات انجام شده روی هر رشته هم بخاطر استفاده از hashTable $O(1)$ است. پس پیچیدگی زمانی هم n^2 است.