



## طراحی الگوریتم‌ها

نیم‌سال دوم ۹۹-۰۰

مدرس: مسعود صدیقین

## تمرین چهارم

## مسئله‌ی ۱\*. سال نو مبارک

اگر  $u$  یک رشته از حروف متوالی  $v$  باشد می‌گوییم  $u$  زیررشته‌ی  $v$  است. مثلاً  $abc$  و  $abd$  به ترتیب زیررشته‌های  $abcd$  است و نیست. حال  $n$  رشته به ما داده شده است و طولانی‌ترین زیررشته‌ی مشترک آن‌ها را باید حساب کنیم. برای این کار الگوریتمی از مرتبه‌ی  $O(nm^2)$  ارائه دهید.

## مسئله‌ی ۲. هفته‌ی آخر ساله

یک مهره‌ی شاه در یک خانه از یک صفحه‌ی مختصات شطرنجی با ابعاد  $n \times n$  قرار دارد. این مهره با احتمال برابر به یکی از هشت خانه‌ی مجاور می‌رود. مهره‌ی شاه دقیقاً  $m$  حرکت می‌کند. الف) الگوریتمی از مرتبه‌ی زمانی  $O(mn^2)$  ارائه دهید که احتمال به اشتباه خارج کردن مهره از صفحه‌ی شطرنج در یکی از نوبت‌ها را بدست آورد. ب) الگوریتمی با کاربرد مشابه و از مرتبه‌ی زمانی  $O(m^3)$  ارائه دهید.

## مسئله‌ی ۳. ای آقا!

فرض کنید آیی  $n$  کاربر در شبکه را گرفته‌اید. الگوریتمی از مرتبه‌ی زمانی خطی ارائه دهید که محاسبه کند به چند روش می‌توانید این آیی‌ها را طوری بین افراد پخش کنید که آیی هیچ‌کس مجدداً به خودش اختصاص نیابد.

## مسئله‌ی ۴\*. فتح دره

فرض کنید ساختاری مثلثی از اعداد به ما داده شده است. (شکل زیر را ببینید) الگوریتمی ارائه دهید که کوتاه‌ترین مسیر (از نظر جمع اعداد)، از قله تا پایین‌ترین سطح مثلث را به ما خروجی دهد. توجه کنید ما در هر مرحله می‌توانیم به یک خانه مجاور پایینی در مثلث حرکت کنیم.

		۲		
	۳		۷	
	۸	۵	۶	
۶	۱	۹	۳	

برای مثال جواب برای مثلث بالا،  $۱۱ = ۲ + ۳ + ۵ + ۱$  است.

## مسئله ۵\*. بزرگترین زیر دنباله مشترک

دو دنباله  $X$  و  $Y$  با طول‌های  $m$  و  $n$  داده شده‌اند. هرکدام از این دو دنباله از مونومرهای  $A, T, G, C$  ساخته شده‌اند. هدف ما پیدا کردن بهترین هم‌ترازی (alignment) بین دو دنباله است. بهترین هم‌ترازی بیشترین امتیاز را طبق طرح زیر دریافت می‌کند:

- $S_{i,j} = 2$  اگر  $X_i$  با  $Y_j$  برابر است. (match score)
- $S_{i,j} = -1$  اگر  $X_i$  با  $Y_j$  برابر نیست. (mismatch score)
- $S_{i,j} = -2$  اگر  $X_i$  یا  $Y_j$  برابر با فضای خالی است. (gap penalty)

برای مثال فرض کنید داشته باشیم:

$X = G A A T T C A G T T A$

$Y = G G A T C G A$

$m = 11, n = 7$

آنگاه یکی از هم‌ترازی‌های بهینه برای این دنباله‌ها به صورت زیر است:

G	A	A	T	T	C	A	G	T	T	A
G	G	A	_	T	C	_	G	_	_	A

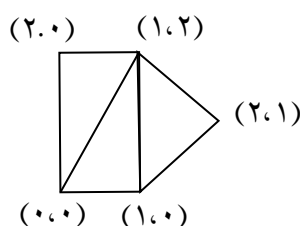
که امتیاز آن به صورت زیر محاسبه می‌شود:

$$S = 2 - 1 + 2 + 2 - 2 + 2 - 2 + 2 - 2 - 2 + 2 = 3$$

## مسئله ۶\*. مثلث‌بندی

یک  $n$  ضلعی محدب داریم و می‌خواهیم آن را مثلث‌بندی کنیم. منظور از یک مثلث‌بندی برای یک  $n$  ضلعی افزایش آن به مثلث‌هایی است که رئوس آن‌ها رئوس  $n$  ضلعی باشند. هزینه‌ی یک مثلث‌بندی مجموع محیط مثلث‌هاست. الگوریتمی از  $O(n^3)$  ارائه کنید که کمترین هزینه را برای مثلث‌بندی پیدا کند.

برای مثال در شکل زیر یک مثلث‌بندی داده شده‌است که در آن دو مثلث با محیط  $3 + \sqrt{5}$  و یک مثلث با محیط  $2 + 2\sqrt{2}$  داریم یعنی هزینه آن  $8 + 2\sqrt{2} + 2\sqrt{5}$  است.



## مسئله ۷\*. اعداد شلخته

به عدد  $a$  شلخته می‌گوییم هرگاه اگر صفرهای سمت چپ عدد را در نظر نگیریم، اختلاف هر دو رقم مجاور آن حداقل دو واحد باشد. برای مثال عدد ۱۵۹۳ عددی شلخته است ولی عدد ۱۵۳۲ شلخته نیست. دو عدد  $l$  و  $r$  به ما ورودی داده می‌شود و می‌خواهیم که اعداد شلخته در بازه  $[l, r]$  را بدست آوریم. اگر اندازه طول عدد  $r$  برابر با  $|r|$  باشد، الگوریتمی ارائه دهید که در زمان  $O(|r|)$  تعداد اعداد شلخته موردنظر را پیدا کند.

## مسئله ۸. پالیندروم

یک پالیندروم، رشته‌ای است که از دو طرف یکسان خوانده می‌شود. اگر اجازه دهیم که یک پالیندروم تنها شامل یک حرف باشد، هر رشته‌ای را می‌توان به صورت دنباله‌ای از پالیندروم‌ها دید. ما می‌خواهیم با داشتن رشته‌ی  $s$  مقدار  $MinPal(s)$  را محاسبه کنیم که برابر است با کمینه مقدار پالیندروم‌هایی که می‌توان با آن‌ها رشته  $s$  را ساخت. یعنی کمترین  $k$  ممکن که  $s$  را بتوان به صورت  $w_1 w_2 \dots w_k$  نوشت به طوری که همه‌ی  $w_1, w_2, \dots, w_k$  پالیندروم باشند. الگوریتمی طراحی کنید که در زمان  $O(n^3)$  که  $n$  طول رشته است، مقدار  $MinPal(s)$  را محاسبه کند.

## مسئله ۹\*. جمع مستطیلی

یک ماتریس  $A_{n \times m}$  از اعداد صحیح داده شده است (مثبت یا منفی) تعریف می‌کنیم:

$$sum(u, v, k, l) := \sum_{i=u}^k \sum_{j=v}^l a_{i,j}, \quad 0 \leq u \leq k \leq n, \quad 0 \leq v \leq l \leq m$$

درواقع تابع  $sum$  محاسبه می‌کند که جمع اعداد در زیرمستطیلی از ماتریس  $A$  که بین دو درایه‌ی  $(u, v)$  و  $(k, l)$  قرار دارد چقدر است. الگوریتمی طراحی کنید که در زمان  $O(nm)$  مقادیر  $u, v, k, l$  را طوری پیدا کند که  $sum(u, v, k, l)$  بیشینه شود.

## مسئله ۱۰\*. احتمالا برای رشته‌ها

یک ماتریس با ابعاد  $m \times n$  داریم. الگوریتمی از مرتبه  $O(m \times n \times k)$  ارائه دهید که دقیقاً با  $k$  دور از گوشه بالا چپ ماتریس به گوشه پایین راست ماتریس برسیم. یک دور زمانی اتفاق می‌افتد که بعد از یک حرکت رو به پایین بلافاصله به راست برویم؛ یا زمانی که بعد از یک حرکت رو به راست بلافاصله به پایین حرکت کنیم. تنها مجاز به استفاده از حرکت‌های راست و پایین هستیم.

## مسئله ۱۱\*. کاربرد تو کجاست دقیقاً؟

$n$  جعبه به شکل مکعب مستطیل در اختیار داریم. می‌خواهیم طوری آن‌ها را روی هم قرار دهیم که بیشترین ارتفاع ممکن را داشته باشیم. دقت کنید که در چینش مکعب‌ها می‌توانیم هر مکعب را هر طور که می‌خواهیم بچرخانیم و ارتفاع آن هر یک از ابعاد آن می‌تواند باشد. یک جعبه را تنها زمانی روی جعبه دیگر می‌توانیم قرار دهیم که ابعاد کف جعبه بالایی از ابعاد متناظر جعبه پایینی کمتر باشد. الگوریتمی از مرتبه  $O(n^2)$  ارائه دهید.

## مسئله ۱۲\*. کاربرد داری ولی به کارم میای؟

یک رشته به طول  $n$  و یک الگو به طول  $m$  داده شده است. می‌خواهیم بدانیم آیا این دو مطابقت دارند یا خیر. الگوریتمی از مرتبه  $O(nm)$  ارائه دهید. در الگو تنها از دو کاراکتر  $*$  و  $*$  و کاراکترهای رشته اصلی استفاده می‌شود. کاراکتر ستاره به معنی صفر، یک یا چند کاراکتر دلخواه از رشته اصلی و کاراکتر علامت سؤال به معنی هر کاراکتر دلخواه از رشته اصلی است.

مثال:

ورودی:

XYXZZXY

X\*\*\*X

خروجی: خیر

ورودی:

XYXZZXY

X\*\*\*X?

خروجی: بله

## مسئله ۱۳\*. کاربرد داری و به کارم میای. خیر ببینی.

دو رشته به ما داده شده و کمترین هزینه تبدیل یکی به دیگری را می‌خواهیم. برای این تبدیل، تنها مجاز به استفاده از این عملیات هستیم. نخست، حذف یک حرف. هزینه حذف هر حرف یک میزان مشخصی برای خودش است و می‌تواند از هزینه حذف هر حرف دیگری متمایز باشد. دوم، اضافه کردن یک حرف. هزینه این عمل نیز مانند حذف، برای هر حرف مقداری مشخص و احتمالا متمایز از سایرین است. سوم، عمل جابجایی. جابجایی دو حرف نیز مقداری احتمالا متمایز از سایر جابجایی‌هاست. با استفاده از این عملیات، الگوریتمی بهینه برای این مسئله ارائه کنید.

پی‌نوشت: تا به حال برایتان پیش آمده که به خاطر نزدیکی ۲ کلید در صفحه‌کلید، کلمه‌ای را اشتباه تایپ کنید؟ :

