



دانشگاه تهران

دانشکدگان فنی

دانشکده علوم مهندسی

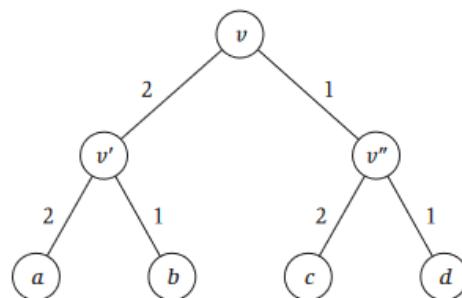
طراحی الگوریتم، پاییز ۱۴۰۳

۸۱۵۸۰۱۰۷۶
سیستمی سرمهباز
۸۱۵۸۰۱۰۷۵
مساکر میان
۸۱۵۸۰۱۰۷۴



تمرین‌های الگوریتم‌های Greedy

- یک الگوریتم کارا ارائه کنید که با دریافت یک مجموعه $\{x_1, x_2, \dots, x_n\}$ از نقاط بر روی خط حقیقی، کوچکترین مجموعه از بازه‌های بسته واحد را تعیین می‌کند که شامل تمام نقاط داده شده در مجموعه باشد. اثبات کنید که الگوریتم شما صحیح است.
- مدارات زمان‌بندی یک جزء حیاتی از تراشه‌های VLSI هستند. در اینجا یک مدل ساده از چنین مداری ارائه شده است. یک درخت دودویی متوازن کامل با n برگ در نظر بگیرید، که n یک توان از دو است. هر یال e از درخت دارای طول l_e است که یک عدد مثبت است. فاصله از ریشه تا یک برگ خاص برابر است با مجموع طول تمامی یال‌های موجود در مسیر از ریشه تا آن برگ. ریشه یک سیگنال ساعت تولید می‌کند که در طول یال‌ها به برگ‌ها منتقل می‌شود. فرض می‌کنیم زمانی که طول می‌کشد تا سیگنال به یک برگ خاص برسد، متناسب با فاصله از ریشه تا آن برگ است. حال اگر تمامی برگ‌ها فاصله یکسانی از ریشه نداشته باشند، سیگنال به طور همزمان به برگ‌ها نمی‌رسد، و این یک مشکل بزرگ است. ما می‌خواهیم که برگ‌ها کاملاً همزمان شوند و همگی در یک زمان سیگنال را دریافت کنند. برای رسیدن به این هدف، باید طول برخی از یال‌ها را افزایش دهیم، به طوری که همه مسیرهای از ریشه به برگ‌ها دارای طول یکسان شوند (ما نمی‌توانیم طول یال‌ها را کاهش دهیم). اگر به این هدف برسیم، گفته می‌شود که درخت (با طول‌های جدید یال‌ها) دارای «صفر شیفت» است. هدف ما رسیدن به «صفر شیفت» به گونه‌ای است که مجموع طول تمام یال‌ها تا حد ممکن کوچک باقی بماند. یک الگوریتم ارائه دهید که طول برخی از یال‌ها را افزایش دهد تا درخت حاصل دارای صفر شیفت باشد و مجموع طول یال‌ها تا حد ممکن کوچک باشد.

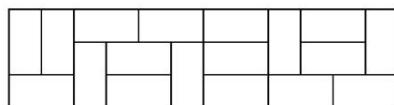


- اخیرا بابک به جمع‌آوری کارت‌های فوتbalی علاقه‌مند شده است. هر نوع کارت یک امتیاز مشخص (از اعداد طبیعی) دارد که آن‌ها را از انواع دیگر متمایز می‌کند. او n کارت از فروشگاه خریداری کرد ولی متوجه شد که متاسفانه در بین آن‌ها کارت‌های مشابه وجود دارد (کمترین امتیاز کارت‌هایی در دست او برابر ۲ است). او علاقه دارد بیشترین تعداد کارت‌های ناهمسان را داشته باشد پس تصمیم گرفت که تعدادی از کارت‌هاییش را با کارت‌هایی دوستانش معاوضه کند. در این معاوضه هیچ محدودیتی وجود ندارد جز اینکه فقط کارت‌هایی می‌توانند معاوضه شوند که اختلاف امتیاز آن‌ها ۱ باشد و همچنین نمی‌تواند کارت‌هایی که در معاوضه به دست آورده را مجدداً معاوضه کند. با یک الگوریتم از مرتبه زمانی $O(n \log n)$ به بابک کمک حداکثر تعداد کارت‌های ناهمسانی که می‌تواند داشته باشد را محاسبه کنند.

۴. فرض کنید مجموعه $S = \{1, 2, \dots, 1000000\}$ را داریم و می‌دانیم که یک زیرمجموعه 101 عضوی از آن به نام A وجود دارد. آیا می‌توان 100 عضو متمایز از S مانند x_1, x_2, \dots, x_{100} پیدا کرد به طوریکه تمامی مجموعه‌های به فرم $A + x_i$ دو به دو مجزا باشند (هیچ دوتایی اشتراک نداشته باشند)? در هر دو صورت برای کسب نمره پاسخ خود را همراه با دلیل منطقی و اثبات محور ارائه دهید.

تمرین‌های الگوریتم‌های DP

۵. بابک که از کاشی‌کاران خوب کشور است به دلیل علاقه شخصی اش به ماهیت علوم مهندسی، به فکر انجام یک تحقیق بین رشته‌ای در حوزه بهینه‌سازی و کاشی‌کاری افتاده است! او که از کاشی‌کاری‌های یکنواخت بازار خسته شده است، می‌خواهد بداند هر زمینی که به فرم $n * 3$ است را به چند روش می‌توان با کاشی‌های به فرم $2 * 1$ یا $1 * 2$ پر کرد؟ به او الگوریتمی ارائه دهید که بتواند این مسئله را حل کند، همچنین الگوریتم خود را تحلیل زمانی کنید.

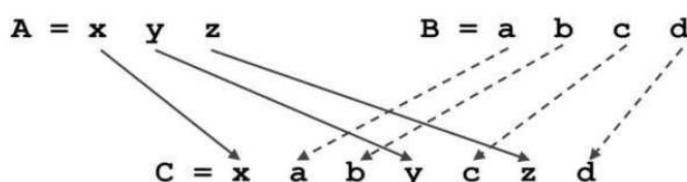


۶. فرض کنید ما یک دایره از N عدد داشته باشیم. یک الگوریتم از $O(N)$ بنویسید که یک بلاک پیوسته روی دایره پیدا کند که بیشترین جمع ممکن را داشته باشد.

۷. گفته می‌شود رشته C در هم‌تنیده‌ی رشته‌های A و B است اگر شامل تمام کاراکترهای A و B باشد و ترتیب نسبی کاراکترهای هر دو رشته در C حفظ شود. به عنوان مثال، اگر مقادیر A, B و C به شرح زیر باشند

$$\begin{aligned} A &= xyz \\ B &= abcd \\ C &= xabcyzd \end{aligned}$$

رشته C در هم‌تنیده‌ی رشته‌های A و B است همان‌طور که در تصویر نشان داده شده است:



با داشتن سه رشته A , B و C . الگوریتمی بنویسید که بررسی کند آیا رشته سوم در هم‌تنیده‌ی دو رشته اول و دوم است یا خیر.

۸. قصد داریم از سرزمینی مستطیلی شکل با m ایالت در هر سطر و n ایالت در هر ستون گذر کنیم. برای عبور از هر خانه باید مقدار مشخصی عوارض بپردازیم. قصد داریم از گوشه بالای سمت راست سرزمین وارد و از گوشه پایین سمت چپ سرزمین خارج شویم و در این حرکت فقط فقط می‌توانیم به چپ، راست و پایین حرکت کنیم و مجاز به حرکت به سمت بالا نیستیم. شبه کدی ارائه کنید که با الگوریتمی بهینه، حداقل پولی که باید برای گذر از این سرزمین همراه داشته باشیم را محاسبه کند.

یک الگوریتم کارا ارائه کنید که با دریافت یک مجموعه $\{x_1, x_2, \dots, x_n\}$ از نقاط بر روی خط حقیقی، کوچکترین مجموعه از بازه‌های بسته واحد را تعیین می‌کند که شامل تمام نقاط داده شده در مجموعه باشد. اثبات کنید که الگوریتم شما صحیح است.

الگوریتم حریمان را به صورت زیر در اینجا دهم: در هر مرحله ترتیب نقطه‌ها که پوشش داده شده (مثلاً x_i) را بپیدار کرد و بازه $[x_i + h, x_i]$ را که پوشش آن انتخاب می‌کنم و آن قدر کار را ادامه دهم تا همه نقاط پوشش داده شوند.

تحلیف زبان: که مرتب سانی اینجا نگاه نیاز به $O(n \lg n)$ زمان دارد، بعد از آن انتخابی را که مرتب کرده و برای اینکه یک نقطه پوشش داده شده باشد را با $O(1)$ انجام دهد که \Leftrightarrow اینجا دهم

اثبات دستی: برای اثبات (رسی)، فرض کنید که برای P از بازه‌ها واحد، بازه‌ها اینها را بازه‌های باشد. فرض کنید جواب بینهای محدود P' هم داریم که در هر دوی اینجا الگوریتم حریمان را باشد. اینجا اینکه P' مطالعه شروع شده اند فرض کنید که اینها بازه‌ها برجسته نقطه شروع شده باشند که بازه‌ها P, P' مغایرند. بنابراین $P \neq P'$.

است: می‌دانیم مبت جیب بازه‌ها اینجا دهم که P' مطالعه شروع شده باشد. اینجا اینکه جواب P بتواند عدد x_i را مطالعه کرده باشد مثل x_i ، بنابراین برای اینکه جواب P بتواند عدد x_i را پوشش دهد باید مطالعه P' قبل از x_i باشد (حرکت تمام نقاط قبل x_i باقی نباشد) اینجا الگوریتم حریمان، تقطیع بازه‌ها (یعنی پوشش) داده شده است.

حال اینجا کنید که این بجای P, P' را دو P جایگزین کنیم، همچنان چهی نقاط توسعه P داشته باشند که بازه‌های اینکه درست می‌باشند قبل بیان شده که چهی نقاط قبل از x_i مطالعه بازه‌ها پوشش داده شده اند، که بازه‌های اینکه درست می‌باشند قبل بیان شده که چهی نقاط قبل از x_i مطالعه بازه‌ها پوشش داده شده اند، واضح است که این P'_n بحث جیب زدن را داده شود، نقطه x_i

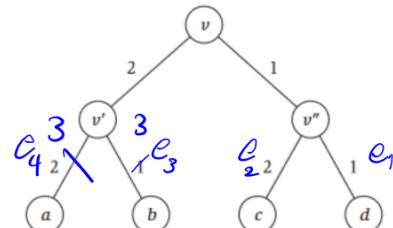
از دست نمی‌رود

بین ترتیب را با کار این عملیات می‌توان نشان داد که تعداد بازه‌ها P و هر جواب بازه‌کشان است و در نتیجه P هم یک جواب بینه است

لم ۱: بایل ها یک رأس به (روجای اس)، در صورتی که بیچه‌ها برگ باشند باید هم وزن باشد.

ابتدا از آنجایی که سیگنال درودی به پر بیچه‌ها باید به برگ‌ها منتفع شود و متوجه که طبق لکس سیگنال به برگ‌ها باید هم باشد (لکسان باشد) می‌توان با نتیجه گرفت که بایل برگ‌ها به پر باید هم وزن

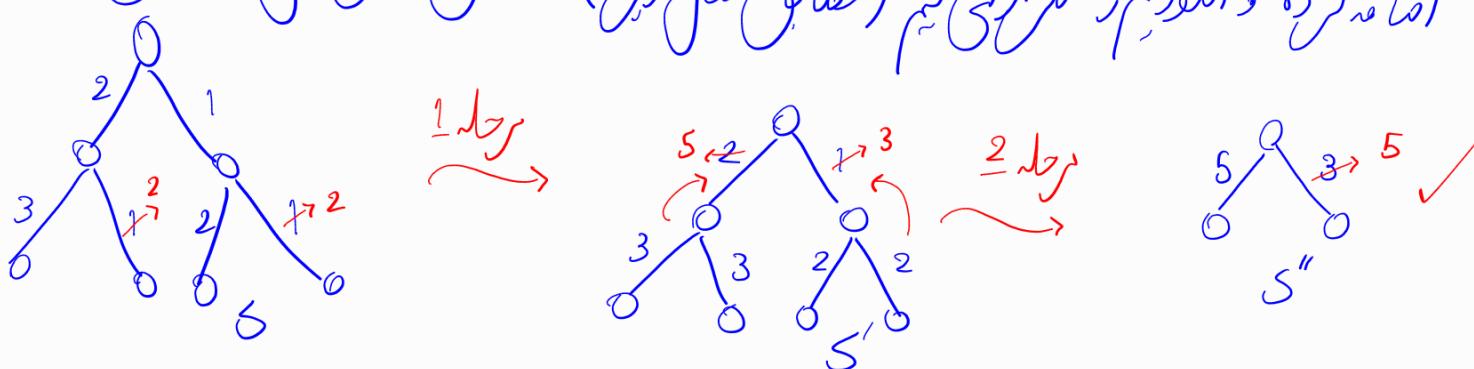
۲. مدارات زمان‌بندی یک جزء حیاتی از تراشه‌های *VLSI* هستند. در اینجا یک مدل ساده از چنین مداری ارائه شده است. یک درخت دودویی متوازن کامل با n برگ در نظر بگیرید، که n یک توان از دو است. هر بایل e از درخت دارای طول e_1 است که یک عدد مثبت است. فاصله از ریشه تا یک برگ خاص برابر است با مجموع طول تمامی بایلهای موجود در مسیر از ریشه تا آن برگ. ریشه یک سیگنال ساعت تولید می‌کند که در طول بایلهای به برگ‌ها منتقل می‌شود. فرض می‌کنیم زمانی که طول می‌کشد تا سیگنال به یک برگ خاص برسد، متناسب با فاصله از ریشه تا آن برگ است. حال اگر تنامی برگ‌ها فاصله یکسانی از ریشه نداشته باشند، سیگنال به طور همزمان به برگ‌ها نمی‌رسد، و این یک مشکل بزرگ است. ما می‌خواهیم که برگ‌ها کاملاً همزمان شوند و همگی در یک زمان سیگنال را دریافت کنند. برای رسیدن به این هدف، باید طول برخی از بایلهای را افزایش دهیم، به طوری که همه مسیرهای از ریشه به برگ‌ها دارای طول یکسان شوند (ما نمی‌توانیم طول بایلهای را کاهش دهیم). اگر به این هدف برسیم، گفته می‌شود که درخت (با طول‌های جدید بایلهای) دارای «صفر شیفت» است. هدف ما رسیدن به «صفر شیفت» به گونه‌ای است که مجموع طول تمام بایلهای تا حد ممکن کوچک باقی بماند. یک الگوریتم ارائه دهد که طول برخی از بایلهای را افزایش دهد تا درخت حاصل دارای صفر شیفت باشد و مجموع طول تمام بایلهای تا حد ممکن کوچک باشد.



$$c_4 = c_3 = e_2 = e_3 \quad \text{and} \quad c_2 = c_1 = e_1$$

حال با استفاده از این لام، الگوریتم که حل مسئله را می‌نماید:

الgoritم مرتبه‌اند: ابتدا با توجه به لام ۱، وزن بایل هر برگ به پریش را در صورت نیاز اضافه کنیم (اضافه کردن هر برگ را حذف کرده و وزن بالا سیگنال را به بایل پریش کنیم) تا وزن همی (سلطان سیگنال زیر) بایل ها منتفع شوند.



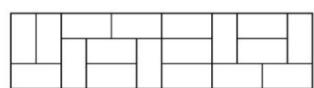
ابتدا درست: با توجه به لام ۱، حداقل هزنه بایل برگ‌ها سخنوار است.

Max وزن بایل در برگ که برداشته شود

حال فرض کنید که درخت جریده کی را از دری درخت از لیه (S) به روی ارازه دهید
الگوریتم ساختم (جنف برگ ها و اضافه کردن وزن شاخص به مال پر) ادعا کیم که باید حساب
بین جریده کی معاملاتی که جواب این بروکار است. اگر این ادعا درست باشد
آنقدر این عمل را (درست مسأله الگوریتم) تکرار کر دو باید حساب بین جریده
بر این اثبات این ادعا به این صفت عملی کنم فرض کنید که جواب بین جریده
طیم کنایه ای است. مطلب این است که این دلیل ها برگ ها آن باشد چون
که طیم کنایه ای از دری (B)، طیف الگوریتم (B) (یعنی مسازم) حال فرض کنید باید حساب
بین جریده کنایه ای. در این صفت جمع مال ها C (که باید حساب بین جریده کنایه ای است)
از B کتر است، حال اگر از مال های پر بینی در C وزن های سائب با مال های برگ ها
در کلم کنیم و در C بینیم (عمل اعمالات الگوریتم را بر عین طبق کنیم) به دفعه می خواهد
که در وجوه در اعمالات رسید (از' B به B، C به C، از' C به C) که جمع مال های هر دو
کنایه ای (اضافه کردن شود) رسید (نشیج) و کنیم که جمع مال های C از B کتر است
که نتایج این سه انجیاب بین جریده (B) بی قول باید حساب بین جریده کنایه ای
کرد. الگوریتم اثبات می شود.

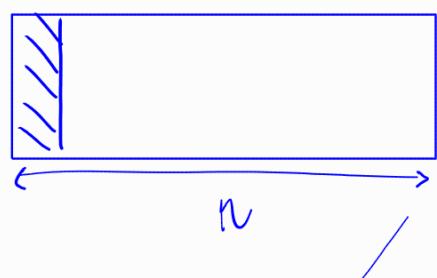
تکمیل (اردرزیانی): درین پیش این درخت (DFS) بر این قاعده که قبل محاسبه کنیم
که اینجا دراد \Rightarrow اردرزیانی $O(2^n) = O(V+E) = O(n)$ است اما (هر مال باید حداقل یکبار
نخورد برگ ها ورودی) لزوم شد درین قاعده از اردرزیانی طبی است!

۵. بایک که از کاشی کاران خوب کشور است به دلیل علاقه شخصی اش به ماهیت علوم مهندسی، به فکر انجام یک تحقیق بین رشته‌ای در حوزه بهینه‌سازی و کاشی کاری افتاده است! او که از کاشی کاری‌های یکنواخت بازار خسته شده است، میخواهد بداند هر زمینی که به فرم $n \times 3$ است را به چند روش میتوان با کاشی‌های به فرم 2×1 یا 1×2 پر کرد؟ به او الگوریتمی ارائه دهد که بتواند این مسئله را حل کند، همچنین الگوریتم خود را تحلیل زمانی کنید.

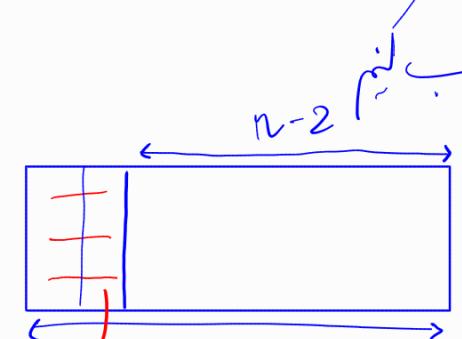


D_n و D'_n نیز می‌توانند حل این مسئله را فراهم کنند:

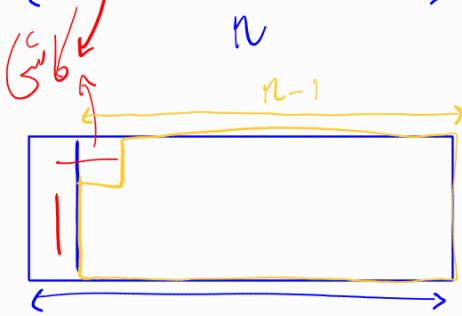
$D_n = 2 \times 1$ با استفاده از کاشی‌ها
 $D'_n = 2 \times 1$ با استفاده از کاشی‌ها



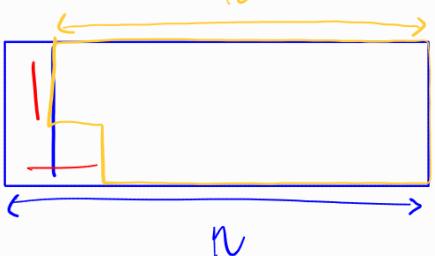
حال را بخطه بازگشتی هر یک از اینها را بیایی کنم
 کاسی را بخطه بازگشتی D_n : به سوی اول جدول
 با هاسهور سفع سه نظری کنم، بنابراین \bar{D}_n



که از \bar{D}_{n-2} حالت زیر را باید بر کاشی کاری (نتیجه کنم)
 (استفاده از ۳ کاشی افقی) که نتیجه D_{n-2} باقی ماند



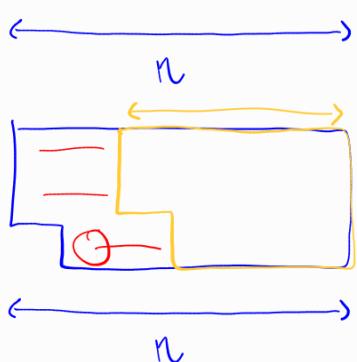
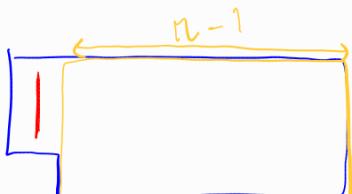
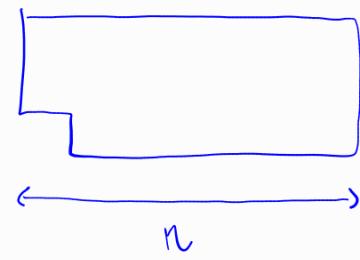
(استفاده از ۱ کاشی افقی) و یک کاشی عمودی که
 دو حالت روی رو را دارد. با توجه به سطح برگسته
 باقی ماند (زد زنگ) D'_{n-1} حالت داریم!



پس از اینکه تولید شیوه گرفت که تعداد روش‌ها
 $D_n = D_{n-2} + 2D'_{n-1}$

بسته باشد

نمودار مکانیکی D'_n : بعداً بازرسی سعین اول



حالات نظریتی

1) سعین اول بازیک کاشی معده برآورد کردار

2) سعین اول بازیک کاشی معده برآورد کردار

2) سعین اول بازیک کاشی معده برآورد کردار

کاخه شکن سند با 0 حدا بازیک کاشی افقی بگذاریم

کردار مدت قدم بازدید و کرایه بگذراند

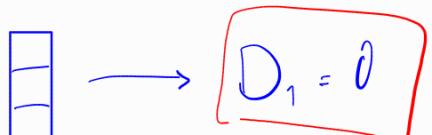
حالات خواهش داشت D'_{n-2}

پنجم: راهنمایی تغییر نسبی رفت

$$D'_n = D'_{n-1} + D'_{n-2}$$

حالات ممکن: بازدید از دستگاه های زیارتی، دستگاه های توزیع آب و برق

$n=1$



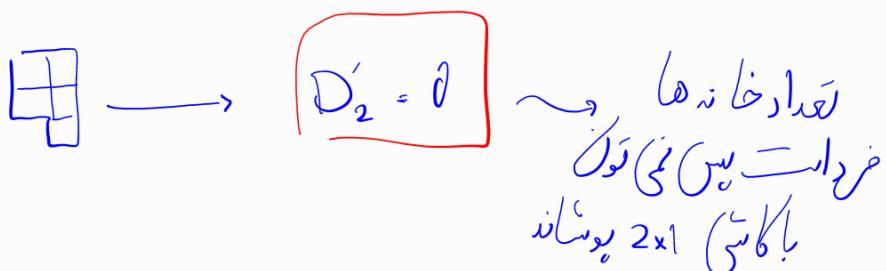
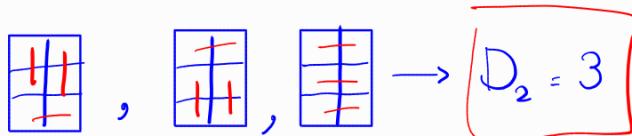
حالات خواهش داشت $D_1 = 0$

تعادل خانه ها



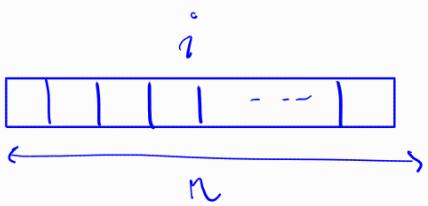
فرداست سوچی کوچه
2x1 بوسان

$n=2$



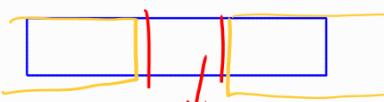
پاتوه به درایلر دست آمده و پایه ها به راحتی می توانند به مدد سکل را باید خواهد بود $O(n)$ کار از for

۶. فرض کنید ما یک دایره از N عدد داشته باشیم. یک الگوریتم از $O(N)$ بنویسید که یک بلاک پیوسته روی دایره پیدا کند که بیشترین جمع ممکن را داشته باشد.


 آنرا سکل را بگیر که عدد روی یک خط مغلق کنیم:
 D_i را حداقل جمع یک بلاک پیوسته نظر نمایی کنیم
 عدد صفت راست آن A_i است (عدد i تا n را باشندگان محسوب نموده بررسی کنیم) اما A_i محسوب نموده باشندگان $i+1$ تا n را باشندگان A_{i+1} است (عدد قبلی i تا n را باشندگان A_{i+1} محسوب نموده که در این صفت جمع آن $= A_i$ است) حالات دارد: ۱) A_i را شامل نموده که در این صفت حداقل جمع A_i است ۲) A_{i+1} را شامل نموده که در این صفت حداقل جمع A_{i+1} است
 $D_{i+1} + A_i$

for D_i باشد $D_0 = A_0$ را صفت داشت $D_i = \max(A_i, D_{i-1} + A_i)$ لیکن با این N حسابی شود. و برای رسیدن جواب بهم باحالات بینی (عو) اینکه صفت راست بازه که این باشد می تواند جواب را محاسبه کرد در اصل $\max_{i=1 \rightarrow N} \{D_i\}$ جواب

خط سکل برگدازه دایره را از یک نقطه دلخواه می شکلیم و به صفت خط درمی آوریم
 خط برگدازه جواب دو حالات داریم: ۱) دورنی زند و به صفت پیوسته روی خط ایم که شود همان جواب که در قسمت قبل نیست آنرا

 ۲) دورنی زند. که در این صفت جواب می شود جمع اندیاد هنوز بازه که از

 جمع را دارد (حرکت که طبقنا بزرگتر است) باشد بازه آن بالکرن
 جمع را حذف کرد) که بزرگ از n حالات نمی درست
 بازه بالکس (جمع

سایه قسمت قبلیک 'O' جدید درستی کلم که به درستی مسابه بازه باکسرن جمع را
لیدای کند که این هم $O(n)$ است.

پس د اینها باک درر بازه باشیش بن جمع و یک در هم بازه باکسرن جمع را حساب
گرده و حواب بی مسود

$\max\{ \text{بازه باکسرن جمع} - \text{جمع مک} \text{ و } \text{بازه باکسرن جمع} \}$

که لهمحی این مسابه هم با در DP ساده و در $O(n)$ درستی آید.

مسایه الگوریتم اراده سده برای LCS در ماس

عملیات کلم. فرخن کند دارم:

$A_i = a_1, a_2, \dots, a_i \rightarrow A$ کارکن از اول i
در نخست این صورت تصریفی کلم

$D_{i,j,k} = \begin{cases} \text{True} & \text{در هم تبعید} \neq A_j \text{ باشد} \\ C_k \end{cases}$
در نخست این صورت

برای ترتیب برای نحوه آیدیت سه این DP هم به صورت زیر می خواهد

را در نظر گیریم:

$D_{i,j,k-1} \leftarrow C_k$ از رئه هاماریمی داشم (1)

$D_{i-1,j,k-1} \leftarrow A$ بعد از در رئه هاماریمی داشم که $C_k = a_i$ در صورتی که $C_k = b_j$ (2)

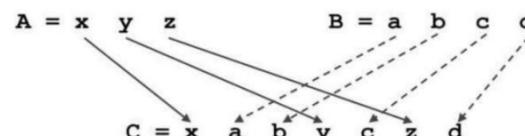
$D_{i,j-1,k-1} \leftarrow "B" \dots \dots \dots C_k = b_j \dots \dots \dots$ (3)

$D_{i,j,k} = \text{of folo} \begin{cases} D_{i,j,k-1} \\ D_{i-1,j,k-1} \quad C_k = a_i \\ D_{i,j-1,k-1} \quad C_k = b_j \end{cases}$ پس با در نظر گرفتن این حالات مکمل برای دارم

۷. گفته می شود رشتہ C در هم تبیین دهی رشتہ های A و B است اگر شامل تمام کاراکترهای A و B باشد و ترتیب نسبی کاراکترهای هر دو رشتہ در C حفظ شود. به عنوان مثال، اگر مقادیر A و B و C به شرح زیر باشند

$$\begin{aligned} A &= xyz \\ B &= abcd \\ C &= xabcyzd \end{aligned}$$

رشته C در هم تبیین دهی رشتہ های A و B است همان طور که در تصویر نشان داده شده است:



با داشتن سه رشتہ A و B . الگوریتمی بنویسید که بررسی کند آیا رشتہ سوم در هم تبیین دهی دو رشتہ اول و دوم است مانند.

که در دستور D_{ijk} با 3 حلقه قدر توجه باشد:

for $i : 0 \rightarrow n$:

 for $j : 0 \rightarrow m$:

 for $k : 0 \rightarrow p$:

i

برای هر کدام از موارد زیر می‌باشد:

$$D_{i,j,k} = \text{False}$$

$$D_{0,0,0} = \text{True}$$

اردد زبان الگوریتم با توجه به 3 حلقه قدر توجه

از $O(nmp)$ اسکالپل از $O(1)$ بحاسی $O(1)$ بعد.

$$A_i = A + x_i$$

فرض کنید مجموعه $S = \{1, 2, \dots, 1000000\}$ را داریم و می‌دانیم که یک زیرمجموعه 101 عضوی از آن به نام A وجود دارد. آیا می‌توان 100 عضو متمایز از S مانند x_1, x_2, \dots, x_{100} پیدا کرد به طوریکه تمامی مجموعه‌های به فرم $A + x_i$ دویده دو مجزا باشند (هیچ دوتایی اشتراک نداشته باشند؟) در هر دو صورت برای کسب نمره پاسخ خود را همراه با دلیل منطقی و اثبات محور ارائه دهید.

آنچه x_i است

علی‌کم ابتدا عدد A شروعی کنید و اضافاً آن را در مجموعه اصلی B از زیرجمله، سپس در مرحله B کوچکترین عدد x_i از A را انتخاب کنید که اضافاً $A + x_i$ با شروع کنید از اضافاً B کوچکترین عدد x_i از $A + x_i$ را اضافاً کرده و این کار را تقدیر ادامه دهد تا 100 عدد

مورد فرض (انتخاب شوند)

حال از انتخاب راه رسمی می‌باشد که این کار برای هر مرحله انجام می‌گیرد اما این کار برای هر مرحله انجام می‌گیرد و وجود دارد. در مرحله اول سه اضافاً $\binom{101}{2} = 5050$ اضافاً کوچکترین عدد x_i را انتخاب کنند.

سی هزار $5050 \cdot 10^6$ عدد دیگر را می‌توان x_i را انتخاب کنند

برای مرحله دوم هر دفعه باید اضافاً A_i باشد که اضافاً قبل از (A, A_1, \dots, A_{i-1}) می‌باشد

که بعد از مرحله دوم هر دفعه باید اضافاً A_2 باشد که اضافاً قبل از (A_1, A_2) می‌باشد

که بعد از مرحله سوم هر دفعه باید اضافاً A_3 باشد که اضافاً قبل از (A_1, A_2, A_3) می‌باشد

مسایل برگزینش
 $10^6 - (5050 + (202 + 303 + 404 + 505 + 606 + 707 + 808 + 909 + 1000) \times 101) = 435\,006$, ۲_{۱۰۱}
 انتخاب خواهیم رانست و از آنجایی که تعداد انتخاب ها n از هم کسر است
 دری با این حال همچنانکه بیشتر از ۱ است پس می توانیم گرفت در هر مرحله قابلیت
 انتخاب $\frac{1}{n}$ را داشته و در نهایت به جواب خواهیم رسید.
 برای تحلیل زمانی الگوریتم هم عملی باید یک آرا به تعداد اعضا کار گرفت که بتوان در $O(n)$
 چک کرد که آن عدد نه، قابل انتخاب است یا نه، برای بررسی ایدار قابل انتخاب
 سرچله هم سمعوی باشد از $O(n)$ مقام سیم (نیما) در رس (اردر نیما)
 است.

۳. اخیراً بابک به جمع آوری کارت‌های فوتbalی علاوه‌مند شده است. هر نوع کارت یک امتیاز مشخص (از اعداد طبیعی) دارد که آن‌ها را از انواع دیگر متمایز می‌کند. او n کارت از فروشگاه خریداری کرد و لی متوجه شد که متسافانه در
 بین آن‌ها کارت‌های مشابه وجود دارد (کمترین امتیاز کارت‌های در دست او برابر ۲ است). او علاقه دارد بیشترین
 تعداد کارت‌های ناهمسان را داشته باشد پس تصمیم گرفت که تعدادی از کارت‌هایی را کارت‌هایی دوستانش معاوضه
 کند. در این معاوضه هیچ محدودیتی وجود ندارد جز اینکه فقط کارت‌هایی می‌توانند معاوضه شوند که اختلاف امتیاز
 آن‌ها ۱ باشد و همچنین نمی‌تواند کارت‌هایی که در معاوضه به دست آورده را مجدداً معاوضه کند. با یک الگوریتم از
 مرتبه زمانی $O(n \log n)$ به بابک کمک کنید حداقل تعداد کارت‌های ناهمسانی که می‌تواند داشته باشد را محاسبه
 کند.

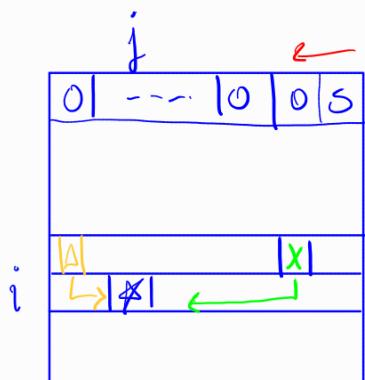
می‌کنیم، در این حال در ساختاری مانند یک دیکشنری (map در C++) داریم که آرای از
 یک امتیاز خاص (کارت) داریم یاده، در هر مرحله رسی هر کارتی که باشیم، چک می‌کنیم آن کارت
 با امتیاز کمتر آن کارت را نداشتم، با آن معاوضه کنیم، در نظر این صورت آن کارتی (دیکشنری)
 هم امتیاز خودش نداشتم، آن کارت را نکوی داریم و در صورتی که مدعی داشتن از آن حال است تبود
 آن را با کارتی با یک امتیاز بیشتر معاوضه کنیم بطور خلاصه، کارت را با کارتی که حداقل
 امتیاز دارد و از آن نداریم معاوضه کرد و به اینکه الگوریتم ادامه داشم مثلاً آن ترازنگاری کارت
 با امتیاز ۴ داشته باشم، اگر را با ۴ معاوضه کنم را نکه داشته و سعی را با ۵ معاوضه کنم.
 جدول چک کردن (راشتر) یا نداشتن یک کارت است، سیمه بفع

بیان داده سازی، سوال این فرض کرد (فرض) که از درجه مطلوب راست
 اثبات (رسخ): فرض کنید که الگوریتم $\text{Greedy}(S)$ نتایجی نباشد، الگوریتم $\text{Greedy}(S)$ را
 می‌دانیم A درجه مطلوب خواهد بود و کارتهای C_i و C_j و فضیلت $\text{swap}(A, C_i, C_j)$ را
 که بازیگر C_i و C_j را بین کارتهای A و B می‌سازد، می‌دانیم $\text{swap}(A, C_i, C_j)$ می‌بدارد.
 داشتن فرض کنید در A دو کارت با این کارتهای C_i و C_j باشند که بازیگر $\text{swap}(A, C_i, C_j)$ را انجام داده باشد. در این صورت یعنی قبل از این
 کارت C_i با کارت C_j با این کارتهای C_i و C_j با این کارتهای C_i و C_j باشند.
 ۱) کارت C_i با کارت C_j با این کارتهای C_i و C_j باشند. در این صورت یعنی در
 کارت C_i با این کارتهای C_i و C_j باشند. این انتخاب را که $\text{swap}(A, C_i, C_j)$ که تعداد کارت ها کم
 شود، می‌دانیم. کارت C_i با این کارتهای C_i و C_j باشند.
 ۲) کارت C_i با کارت C_j با این کارتهای C_i و C_j باشند. در این صورت یعنی در
 ۱-۱: الگوریتم Greedy کارت C_i با این کارتهای C_i و C_j باشند. که $\text{swap}(A, C_i, C_j)$ کارت
 با این کارتهای C_i و C_j باشند. با تغییر انتخاب در $A \leftarrow A - C_i + C_j$ تعداد کارت ها کم شود،
 پس این کارت را نتیجه گیریم.
 ۱-۲: الگوریتم Greedy کارت C_i با این کارتهای C_i و C_j باشند.
 نتیجه گرفت که کارت C_i با این کارتهای C_i و C_j باشند. پس با تغییر انتخاب $A \leftarrow A - C_i + C_j$
 تعداد کارت ها کم شود، پس این کارت را نتیجه گیریم.
 ۳) کارت C_i با کارت A با کارت C_j با این کارتهای C_i و C_j باشند. در این صورت، الگوریتم Greedy کارت
 C_i با کارت A با این کارتهای C_i و C_j باشند. که $\text{swap}(A, C_i, C_j)$ کارت A با این کارتهای C_i و C_j باشند.

انتخاب A را با انتخاب الگوریتم حریم‌ساز معروف کنیم تعداد کارهای کمتر که انتخاب کنم می‌شود
که بتوحیث استین که با آن در هر جایی که الگوریتم A با قدرت داشته باشد می‌توان انتخاب A
را بدینکه تعداد کارهای کم شود و بجز ترتیب با تکرار این عملیات
و فرآیند A را بدینکه تعداد کارهای کم شود و بجز ترتیب با تکرار این عملیات

$D_{i,j}$ را نظریه‌ی کنم: هزینه کارهای
برکار رسیدن خانه i در j (در مطری)

قصد داریم از سرمیمی مستطیلی شکل با m ایالت در هر سطر و n ایالت در هر ستون گذر کنیم. برای عبور از هر خانه باید مقدار مشخصی عوارض پیردازیم. قصد داریم از گوشه بالای سمت راست سرمیمین وارد و از گوشه پایین سمت چپ سرمیمین خارج شویم و در این حرکت فقط می‌توانیم به چپ، راست و پایین حرکت کنیم و مجاز به حرکت به سمت بالا نیستیم. شبیه کدی ارائه کنید که با الگوریتمی بهینه، حداقل پولی که باید برای گذر از این سرمیمین همراه داشته باشیم را محاسبه کند.



و سفر (j, m) باید پرداخت کنم
محاسبه رابطه بازگشتی: برکار رسیدن به خانه‌ها سفر اول که با
داری مسخر (شده‌اند)، واضح است که حداقل هزینه با حرکت درست
قطعه خرمن زنگ نیست و آنرا پس حالت پایه مانع شود $D_{0,0}$

برکار رسیدن رابطه بازگشتی مسیر خانه‌ها روند اینکه در مسیر پیش از اینکه از این خانه به سفر نمایم احتمالات اینست که کم، حرکت اگر این خانه مسخر (باشد) لغایه مسیر پیش از این خانه مسخر (بتوشود) می‌گذرد اگر از خانه X به \neq بایم مسیر پیش از خانه مسیر خانه از خانه

بسیار بسیار ساده می‌شود (حرکت برکار چپ و راست محدودیت نداریم و)
برکار پایانی مسیر پیش از خانه که پایانی (روز) مسیر را تعیین می‌کند).

$$D_{i,j} = \min_{k=0 \rightarrow m} \{ D_{i-1,k} + (i,k) : (i,k) \in \text{مسیر} \}$$

لطفاً ملاحظة أن حجم المربع هو $O(nm)$ ، لذا حجم المربع هو $O(nm^2)$ ، ونحو ذلك

$C[i][j] = \min_{k=1}^{n-m+1} (C[i][k] + C[k][j])$

// حجم المربع

Sum $\rightarrow 0$ اول ثم $n \times m$ ثم T

for i: 1 $\rightarrow n$

Sum[i][0] = C[i][0]

for j: 2 $\rightarrow m$:

Sum[i][j] = Sum[i][j-1] + C[i][j] $\leftarrow O(nm^2)$ $\leftarrow O(V+E)$

D \rightarrow مجموع المربعات

// حفظ الحالات السابقة

for i: 2 $\rightarrow n$:

D[i][0] = Sum[n][0] - Sum[i-1][0]

D[1][0] = Sum[n][0]

// DP (Recursion)

for i: 2 $\rightarrow n$:

for j: 1 $\rightarrow m$:

Min = $+\infty$

for k: 1 $\rightarrow m$:

Min = $\min(\text{Min}, D[i-1][k] + \text{Sum}[i][k] - \text{Sum}[i][j-1] + C[i][k])$

D[i][j] = Min

Print(D[n][m])

أولاً

ثانياً

الآن نعم ، لأن المربع

لذلك نعرف ذلك

لذلك نعرف ذلك

لذلك نعرف ذلك

لذلك نعرف ذلك

(Dynamic Programming) (مقدمة في ديناميك البرمجة)

! مراجعة (جواب)

