

ساختمان داده و الگوریتم ها (CE203)

جلسه بیست و سوم:
حل تمرین

سجاد شیرعلی شمرضا

پاییز 1400

شنبه، 4 دی 1400

نمونه سوال از برنامه نویسی پویا

ماراتن تمرین ها

- روزهای آخر ترم همیشه برای زینب مملو از تمرین و پروژه است. او در حال حاضر که k روز از ترم باقی مانده است، می بایست n تمرین را نوشته و ارسال کند. او می بایست که تمرین ها را به ترتیب نوشته و ارسال کند. همچنین او می داند که برای انجام دادن هر تمرین، نیاز به مصرف d_i کالری انرژی دارد. از طرفی، برای اینکه بتواند به راحتی بخوابد، نمی خواهد که هیچ تمرینی را در انتهای روز، نیمه کاره رها کند. به عبارت دیگر، اگر تمرینی را در یک روز آغاز کرد، باید در آن روز نیز تمام کند. همچنین زینب می خواهد که فشاری که بر او وارد می شود، در طی روزهای ممکن تقریباً یکسان باشد. به همین خاطر، او می خواهد به گونه ای برنامه ریزی کند که حداکثر کالری مصرفی برای حل تمرین در هر روز (که مجموع کالری لازم برای تمرین های حل شده در آن روز است)، کمینه شود. برای او یک الگوریتم با زمان اجرای $O(kn^2)$ برای برنامه ریزی چگونگی انجام تمرین ها طراحی کنید.

خلاصه مسئله

- نیاز به حل n تمرین
- تمرین i ام نیاز به d_i کیلو کالری انرژی
- تمرین ها باید به ترتیب حل شوند
- نمی توان در یک روز بخشی از یک تمرین را انجام داد
- کل تمرین ها باید در k روز انجام شوند
- ایده؟

تعریف زیر مسئله

- حل i تدرین اول در k' روز به گونه ای که حداکثر انرژی مصرفی روزانه، کمینه باشد:

$$x(i, k')$$

- مسئله اصلی: حل n تدرین اول در k روز یعنی $x(n, k)$

چگونگی حل زیر مسئله

- مشخص کردن تمرین هایی که در روز آخر حل خواهد کرد
- فرض کنید که در روز آخر، از تمرین $i'+1$ تا i را حل خواهد کرد
- به عبارت دیگر، در $k-1$ روز اول، i' تمرین اول و در روز آخر، بقیه تمرین ها را حل خواهد کرد
- حداکثر انرژی مصرفی روزانه: $\max \left(x(i', k' - 1), \sum_{j=i'+1}^i d_j \right)$
- حداکثر انرژی مصرفی روزانه در $k-1$ روز اول و انرژی مصرفی در روز k ام
- برای کمینه کردن این جواب، باید بهترین انتخاب برای مجموعه تمرین های روز آخر را داشته باشیم، یعنی

$$x(i, k') = \min \left\{ \max \left(x(i', k' - 1), \sum_{j=i'+1}^i d_j \right) \mid i' \in \{0, \dots, i\} \right\}$$

حالت پایه برای زیر مسئله ها

- حل هیچ تمرینی در k' روز نیاز به هیچ انرژی ندارد!

$$x(0, k') = 0 \text{ for } k' \geq 0$$

- اگر روزی باقی نمانده است، حل تمرین ها ممکن نیست (انرژی بینهایت می خواهد)!

$$x(i, 0) = \infty \text{ for } i > 0$$

تحلیل زمان اجرا

- عبارت مورد نظر برای محاسبه مقدار هر زیر مسئله:

$$x(i, k') = \min \left\{ \max \left(x(i', k' - 1), \sum_{j=i'+1}^i d_j \right) \mid i' \in \{0, \dots, i\} \right\}$$

- زمان مورد نظر برای محاسبه این برای یک زیر مسئله:
 - در صورت محاسبه انرژی روز آخر برای هر حالت به صورت جداگانه: $\Theta(i^2) \subset O(n^2)$
 - اگر مجموع انرژی لازم برای تمرین های i تا j قبلا محاسبه شده است: $O(n)$

■ چگونگی در صفحه بعد!

- تعداد زیر مسئله ها: $(n + 1)(1 + k) = \Theta(nk)$

$$O(kn^2)$$

- زمان برای حل تمام زیر مسئله ها:

محاسبه انرژی لازم برای مجموعه ای از تمرین ها

- نمایش انرژی لازم برای حل مجموعه ای از تمرین ها (از تمرین i' تا تمرین i):

$$s(i', i) = \sum_{j=i'+1}^i d_j$$

- رابطه ای بازگشتی برای محاسبه: $s(i', i) = d_i + s(i', i - 1)$

- حالت پایه: $s(i', i') = 0$

- تعداد زیر مسئله ها: $O(n^2)$

- زمان محاسبه تمام مقادیر (با توجه به نیاز به زمان ثابت برای محاسبه هریک): $O(n^2)$



سوال؟

نمونه سوالی دیگر از برنامه نویسی پویا

بهترین حالت انتخاب گروه

- مسئله انتخاب بهترین گروه درس با توجه به محدودیت گروه ها، یکی از دغدغه های بزرگ دانشجویان است. ترم آینده قرار است که $3n$ دانشجو درس ساختمان داده ها و الگوریتم ها را بگیرند و سه استاد (احسان، سجاد، و علیرضا) قرار است سه گروه n ارائه دهند. دانشکده تصمیم گرفته است که بهترین روش ثبت نام دانشجویان در گروه ها را پیدا کند و برای همین از شما کمک خواسته است. دانشگاه از هر دانشجو پرسیده است که در صورت گرفتن درس با هر یک از این اساتید، چقدر مطلب جدید یاد خواهد گرفت. جواب هر دانشجوی i ام به صورت 3 عدد غیر منفی a_i, b_i, c_i که معرف مقدار یادگیری در صورت ثبت نام در گروه احسان، سجاد، و علیرضا است. سوال اول دانشکده این است که آیا می توان افراد را به گونه ای در کلاس ها ثبت نام کرد که میزان یادگیری هر فرد، مثبت باشد (و صفر نباشد)؟ و اگر بله، بهترین نحوه ثبت نام که مجموع میزان یادگیری افراد را بیشینه می کند، کدام است. الگوریتم شما باید از زمان $O(n^3)$ باشد.

خلاصه کردن مسئله

- تعداد دانشجویان: $3n$
- تعداد گروه: 3 (نمایش گروه ها با a و b و c)
- تعداد دانشجو در هر گروه: n
- جواب بهینه: اگر دانشجوی i در گروه x ثبت نام شده است، باید اولاً x_i مثبت باشد، ثانياً مجموع این اعداد برای تمام دانشجویان، بیشینه باشد

ایده برای حل جواب



تعریف زیر مسئله

- مسئله اصلی: ثبت نام هر $3n$ دانشجو در 3 گروه به تعداد مساوی
- زیر مسئله: ثبت نام $i+j+k$ دانشجوی اول به صورتی که در گروه اول، i دانشجو، در گروه دوم، j دانشجو، و در گروه سوم، k دانشجو باشند
 - شرط جواب بهینه همان است (همه مثبت و مجموع بیشینه)
- نمایش جواب بهینه این مسئله به صورت $x(i,j,k)$
 - نمایش به صورت $-\infty$ اگر غیر ممکن است
- مسئله اصلی معادل است با $x(n,n,n)$

رابطه بازگشتی برای زیر مسئله

- دانشجوی آخر (دانشجوی شماره $i+j+k$) باید در یک کلاسی ثبت نام بشود!

$$x(i, j, k) = \max\{-\infty\} \cup \left\{ \begin{array}{ll} a_{i+j+k} + x(i-1, j, k) & \text{if } i > 0 \text{ and } a_{i+j+k} > 0, \\ b_{i+j+k} + x(i, j-1, k) & \text{if } j > 0 \text{ and } b_{i+j+k} > 0, \\ c_{i+j+k} + x(i, j, k-1) & \text{if } k > 0 \text{ and } c_{i+j+k} > 0 \end{array} \right\}$$

$$x(0, 0, 0) = 0$$

- حالت پایه:

تحلیل زمانی

- هریک از i و j و k می توانند عددی بین 0 تا n باشند
- هریک $n+1$ حالت دارند
- تعداد کل زیر مسئله ها: $(n+1)^3 = O(n^3)$
- زمان لازم برای حل هر زیر مسئله: $O(1)$
- کل زمان اجرا: $O(n^3)$



سوال؟