ساختمان داده و الگوریتم ها (CE203)

جلسه بیست و دوم:
حل تمرین

سجاد شیرعلی شهرضا

پاییز 1400

شنبه، 27 آذر 1400

نمونه سوال از مرتب سازی

مسئله پیدا کردن با ارزش ترین کتاب

- جواد دانشجوی سال آخر است. او از ابتدای دوره کارشناسی، کتاب های درسی را که خریده است، به ترتیب از 1 تا n شماره گذاری کرده است. در اسباب کشی اخیر او، بیش از نیمی از کتابهایش در راه گم شده است. از طرفی، از نظر جواد، ارزش هر کتاب ترکیبی از شماره آن کتاب و تعداد صفحات آن کتاب است: $v_i = 3^{p_i}(n-i)^2$ در این رابطه، i شماره کتاب و p_i تعداد صفحات کتاب است. فرض کنید که حداکثر تعداد صفحات کتابها 23lgn است. او می خواهد میانه ارزش کتاب های گمشده را محاسبه کنید. به او در طراحی الگوریتمی برای این کار کمک کنید.

خلاصه کردن مسئله

- تعداد کتاب های گمشده اگر k باشد: $n/2 < k < n$
- شماره هر کدام از کتاب ها : $n+1>i>0$
- تعداد صفحات هر کتاب : $p_i<23\lg n$
- ارزش کتاب iام: $v_i = 3^{p_i}(n-i)^2$
- هدف: پیدا کردن میانه ارزش کتاب های گمشده
- ایده؟

کران پایین برای جواب

؟

کران پایین برای جواب

- نیاز به بررسی تمام کتاب های گمشده
- تعداد کتاب های گمشده اگر k باشد: $n/2 < k < n$
- حداقل زمان لازم:

$$\Omega(n)$$

گام اول

- فرض: مقدار ارزش هر کتاب حساب شده است
- تعداد مقادیر ارزش کتابها: $O(n)$
- تعداد صفحات هر کتاب : $p_i<23\lg n$
- شماره هر کدام از کتاب ها : $n+1>i>0$
- ارزش هر کتاب یک عدد صحیح است
- حداکثر ارزش هر کتاب: $v_i \le 3^{23\lg n} n^2 = O(n^c)$
- امکان مرتب سازی ارزش ها با مرتب سازی مبنایی در زمان $O(n)$

محاسبه ارزش کتاب های گمشده

- ارزش هر کتاب: $v_i = 3^{p_i}(n-i)^2$
- محاسبه ارزش هر کتاب: $\Theta(\log n)$
- محاسبه ارزش تمام کتاب ها: $O(n \log n)$

راه بهتر محاسبه ارزش کتّاب های گَمشده

- تعداد صفحات هر کتاب : $p_i<23\lg n$
- محاسبه 3^{p_i} برای تمام مقادیر ممکن p_i
  - زمان لازم برای محاسبه همه: $O(\log n)$
    - تعداد اعداد مورد محاسبه lgn است و هر کدام از روی قبلی در زمان ثابت محاسبه می شود
- تعیین ارزش هر کتاب در زمان ثابت از روی این مقادیر با فرمول $v_i = 3^{p_i}(n-i)^2$
- زمان محاسبه تمام ارزش ها: $O(n)$

سوال؟

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 کیلو کالری انرژی دارد. از طرفی، برای اینکه بتواند به راحتی بخواهد، 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 تقریبا 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 iام 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 تمرین اول در 'k روز به گونه ای که حداکثر انرژی مصرفی روزانه، کمینه باشد:

$$x(i, k')$$

- مسئله اصلی: حل n تمرین 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 kام
- برای کمینه کردن این جواب، باید بهترین انتخاب برای مجموعه تمرین های روز آخر را داشته باشیم، یعنی

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

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:

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

- 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: $(n+1)(1+k) = \Theta(nk)$
- زمان برای حل تمام زیر مسئله ها: $O(kn^2)$

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 هر یک): $O(n^2)$

سوال؟