

تمرین چهارم طراحی الگوریتم‌ها

اشکان شکيبا (۹۹۳۱۰۳۰)

سوال اول

با توجه به شرایط مسئله و ارتباط و تاثیر مراحل بر هم، واضح است که با روش‌هایی نظیر روش حریصانه نمی‌توان آن را حل کرد و نیاز به یک راه حل از نوع برنامه‌نویسی پویا داریم.

راه حل پیشنهادی به این صورت است که با تعریف جدولی، درایه z, i, m آن متناظر با کمترین هزینه ممکن برای ماشین اول به نقطه i و ماشین دوم به نقطه z باشد. (با در نظر گرفتن سرکشی شدن نقاط کوچک‌تر)

واضح است که درایه‌های $0, 1$ و $0, 1$ مقداری برابر فاصله مبدا تا اولین نقطه خواهند داشت.

روش محاسبه سایر درایه‌های z, i دارای i بزرگ‌تر یا برابر z ، به این صورت است که کمترین هزینه بین درایه $z, i-1$ و درایه‌های i, k (که $k < z$) انتخاب می‌شود.

به همین شکل در انتها پاسخ نهایی برابر کمینه درایه‌های i, n و n, i خواهد بود.

پیچیدگی زمانی الگوریتم $O(n^3)$ است.

سوال دوم

فرض می‌کنیم دو جعبه داشته باشیم. در این صورت می‌توان تا جای ممکن کیسه‌ها را در جعبه اول قرار داد و در انتها مقداری فضای خالی در جعبه می‌ماند و به سراغ جعبه دوم می‌رویم. مشابه‌ها در جعبه دوم نیز ممکن است مقداری فضای خالی بماند. واضح است که جمع این دو فضا همواره کمتر از کل فضای یک جعبه خواهد بود. اگر فضای هر جعبه را یک واحد گنجایش در نظر بگیریم، دو جعبه دارای دو واحد گنجایش خواهند بود که در بدترین حالت، در هر یک تنها یک کیسه قرار می‌گیرد و مجموع کیسه‌ها از یک واحد بیشتر شده و مجموع فضاهای خالی کمتر از یک واحد می‌شود. به طور کلی از این موضوع می‌توان نتیجه‌گیری کرد که حداکثر دو برابر حالت بهینه نیاز به ممکن است داشته باشیم.

سوال سوم

در یک راه‌حل بر پایه برنامه‌نویسی پویا، می‌توان جدولی ۳ بعدی با ابعاد تعداد مکعب‌ها داشت که در هر درایه آن عدد ۶ قرار دارد. در ادامه به ازای انتخاب هر یک از وجوه به عنوان وجه پایین‌تر، یک عدد ثبت می‌شود. مقدار درایه i, j, k نمایان‌گر بیشینه ارتفاع برج در صورت قرار دادن مکعب z بر روی مکعب i از وجه k آن خواهد بود. در هر مرحله بیشترین ارتفاع مرحله پیشین که سازگار باشد انتخاب می‌شود. واضح است که هنگام پیش‌برد الگوریتم و تغییر محتوای درایه‌ها، باید شرط سازگاری رنگ‌ها نیز بررسی شود.