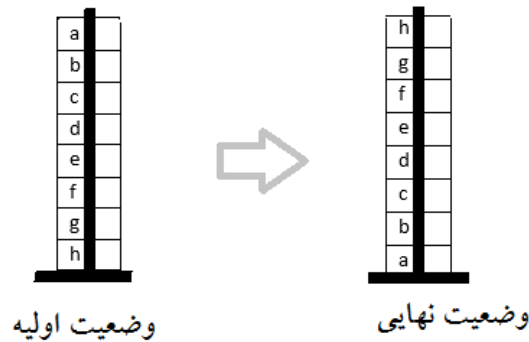


۱. وارون کردن محتویات برج هانوی. اینجا برخلاف مسئله برج هانوی، دیسکها اندازه ندارند و هر دیسک فقط یک شماره منحصر بفرد دارد. دیسکهای با شماره های مختلف را می توان روی هم قرار داد. می خواهیم با کمترین میزان جابجایی محتوای یک برج هانوی وارون شود.



این کار را می خواهیم با داشتن محدودیتهای مختلف انجام دهیم. برای هر یکی از موارد زیر یک الگوریتم ارائه دهید و زمان اجرای آن را ذکر کنید. زمان اجرا برابر با جابجایی هایی است که انجام می دهید.

- (آ) مجاز هستیم از یک برج هانوی دیگر و به تعداد $O(1)$ دیسک روی زمین قرار دهیم.
- (ب) مجاز هستیم از یک برج هانوی دیگر و به تعداد k دیسک روی زمین قرار دهیم.
- (ج) مجاز هستیم از دو برج هانوی دیگر و به تعداد $O(1)$ دیسک روی زمین قرار دهیم.

۲. گزاره های زیر را در مورد درخت باینری ثابت کنید. n_E تعداد برگهای درخت و n_I تعداد رئوس غیر برگ است.

1. $h + 1 \leq n \leq 2^{h+1} - 1$
2. $1 \leq n_E \leq 2^h$
3. $h \leq n_I \leq 2^h - 1$
4. $\log(n + 1) - 1 \leq h \leq n - 1$

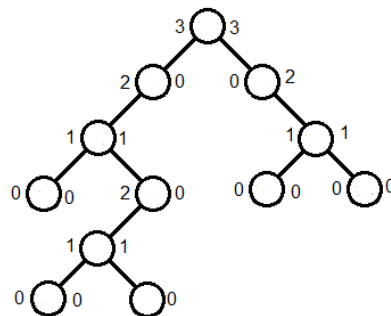
۳. در یک درخت دودویی کامل، هر راس یا فرزند ندارد یا دو فرزند دارد. نشان دهید تعداد رئوس در یک درخت باینری کامل فرد است.

۴. توضیح دهید که n عدد صحیح که از بازه $[1, 100n]$ هستند را چگونه می توان در زمان $O(n)$ مرتب کرد؟ چرا این با کران پایین $\Omega(n \log n)$ برای مرتب سازی در تناقض نیست؟

۵. فرض کنید در هرم بیشینه فرزند سمت چپ دختر و فرزند سمت راست پسر باشد. در نمایش هرم با آرایه، عنصری که در اندیس i ام ذخیره شده، پسر دخترش در چه اندیسی ذخیره شده؟ اگر ریشه دختر باشد، در یک بیشینه با n راس حداقل چند دختر وجود دارد؟

۶. کوچکترین عنصر در یک هرم بیشینه را در چه زمانی می توان پیدا کرد؟ چرا؟ k امین بزرگترین عنصر را در چه زمانی می توان پیدا کرد؟

۷. درخت دودویی T شامل n راس داده شده است. شیب چپ راس x برابر با تعداد قدمهایی است که از x میتوانیم به سمت پایین برویم به شرطی که هر بار به سمت چپ برویم. شیب راست x هم به همین منوال قابل تعریف است. یعنی تعداد دفعاتی که با شروع از راس x میتوانیم به سمت راست برویم. در شکل زیر، شیب چپ و شیب راست هر راس در کنار آن نوشته شده است.



(آ) الگوریتمی طراحی کنید که شیب چپ و راست هر راس درخت را محاسبه کند. زمان اجرای الگوریتم شما چقدر است؟

(ب) می خواهیم ساختار داده ای طراحی و پیاده سازی کنیم که سه عمل اصلی داشته باشد.

i. اضافه کردن ریشه در صورتی که درخت تهی باشد.

ii. یک راس با بیشترین شیب چپ یا راست را گزارش کند. همراه با مقدار شیب

iii. اضافه کردن یک برگ به درخت. فرض بر این است، که آدرس راس x را میدانیم و می خواهیم یک فرزند چپ یا راست به x اضافه کنیم. در صورتی که فرزند قبلا موجود باشد، پیام خطا چاپ شود.

(ج) در راه حل شما زمان اجرای هر یک از اعمال بالا چقدر است؟