



مسئله‌ی ۱*. زمان تنگه

پیچیدگی توابع بازگشتی زیر را تحلیل کنید.

الف) $T(n) = T(\frac{n}{4}) + \frac{n}{\log n}$

ب) $T(n) = 2T(\frac{n}{4}) + 1$

ج) $T(n) = 2T(\frac{n}{4}) + \sqrt{n}$

(محمد مهدی ابوترابی)

مسئله‌ی ۲*. شاخ

یک آرایه یک بعدی را در نظر بگیرید و فرض کنید اعدادی طبیعی در خانه‌های آن وجود دارند. حال می‌خواهیم عضوی از این آرایه را پیدا کنیم که از دو عضو مجاورش بزرگتر است. الگوریتمی را در نظر بگیرید که هر بار عضو میانی الگوریتم را انتخاب کرده و آن را با دو عضو کناریش مقایسه می‌کنیم. سپس اگر آن عضو جواب مورد نظر نبود، به دنبال جواب در دو تکه باقی‌مانده آرایه می‌گردیم. این الگوریتم را با نوشتن رابطه‌ی بازگشتی از نظر زمانی تحلیل کنید.

(علی عباسی)

مسئله‌ی ۳*. اکثریت اقلیت

می‌خواهیم عضو اکثریت یک آرایه را (عضوی که بیش از نصف اعضای یک آرایه را تشکیل می‌دهد)، به دست آوریم. ابتدا اعداد درون یک آرایه را به دو بخش تقسیم می‌کنیم. می‌دانیم عضو اکثریت کل باید در یکی از این دو بخش عضو اکثریت باشد. سپس با مقایسه دو عضو اکثریت این دو بازه و تعداد تکرار آن‌ها در کل آرایه عضو اکثریت کل به دست می‌آید. حال از نظر زمانی الگوریتم مورد نظر را با استفاده از روابط بازگشتی محاسبه کنید.

(علی عباسی)

مسئله‌ی ۴. اثباتی‌فای

حالت سوم از قضیه اصلی را به رسم کمک درخت بازگشت اثبات کنید. برای راحتی فرض کنید n یک توان صحیح از b است.

مسئله ۵*. پوینت گر

در زبان‌های برنامه‌نویسی می‌توان سه استراتژی را برای مواقعی که یک آرایه به عنوان پارامتر به تابعی ارسال می‌شود لحاظ کرد:

۱. آرایه به وسیله اشاره گر ارسال می‌شود: زمان $\Theta(1)$

۲. آرایه به وسیله کپی کردن ارسال می‌شود: زمان $\Theta(N)$ که N طول آرایه است.

۳. آرایه به وسیله کپی کردن تنها زیربازه‌ای که ممکن است توسط تابع فراخوانی شده مورد دسترسی قرار گیرد، ارسال می‌شود. اگر زیرآرایه $A[p..q]$ ارسال شود: زمان $\Theta(q - p + 1)$

الف) الگوریتم جستجوی دودویی بازگشتی برای یافتن یک عدد در آرایه مرتب‌شده را در نظر بگیرید. رابطه‌های بازگشتی برای زمان اجرا در بدترین حالت را، هنگامی که آرایه با استفاده از هرکدام از سه روش فوق ارسال می‌شود، ارائه دهید و سپس به کمک قضیه اصلی آن روابط را حل کنید.

ب) بخش الف را برای الگوریتم مرتب‌سازی ادغامی مجدداً انجام دهید.

(علی عباسی)

مسئله ۶*. سیمبا

سیمبا می‌خواهد یک الگوریتم جدید برای ضرب ماتریس‌ها ارائه دهد که از الگوریتم استراسن را سریع‌تر عمل کند. الگوریتم سیمبا به اینگونه است که هر ماتریس را به بخش‌هایی از سایز $\frac{n}{4} \times \frac{n}{4}$ تقسیم می‌کند و همچنین مراحل تقسیم و ترکیب کردن در مجموع زمانی به اندازه $\theta(n^2)$ می‌گیرد. سیمبا می‌خواهد بداند که الگوریتمش باید چه تعداد زیر مساله ایجاد کند تا از الگوریتم استراسن بهتر عمل کند.

(محمد مهدی ابوترابی)