

سری اول تمرینات درس طراحی و تحلیل الگوریتمها

1 نمادگذاری مجانبی و تحلیل الگوریتمها

۱. اثبات یا رد کنید. گزاره‌های درست را اثبات کنید و برای رد گزاره‌های غلط مثال نقض ارائه دهید.

الف. اگر $f(n)$ و $g(n)$ دو تابع مثبت باشند آنگاه،

$$O(f(n) + g(n)) = f(n) + O(g(n)).$$

ب. اگر $f(n)$ و $g(n)$ دو تابع مثبت باشند آنگاه،

$$f(n) + g(n) = \Theta(\max\{f(n), g(n)\}).$$

ج. اگر $f_1(n) = \Theta(g_1(n))$ و $f_2(n) = \Theta(g_2(n))$ آنگاه،

$$f_2(f_1(n)) = \Theta(g_2(g_1(n))).$$

د. اگر $f(n)$ و $g(n)$ دو تابع مثبت باشند آنگاه، همواره یکی از دو حالت زیر برقرار است:

$$f(n) = O(g(n)) \quad \text{OR} \quad g(n) = O(f(n)).$$

۲. مرتب‌سازی سریع با کمک بر زدن. فرض کنید الگوریتمی به نام Random_Permute وجود دارد که در زمان $O(n \log n)$ یک جایگشت n -تایی تصادفی با توزیع یکنواخت ارائه می‌کند. همچنین الگوریتم مرتب‌سازی سریع با یک استراتژی دلخواه مستقل از ورودی را در نظر بگیرید. نشان دهید اگر به وسیله الگوریتم اول ورودی را بر بزنیم و سپس مرتب‌سازی سریع را اعمال کنیم، زمان اجرای مورد انتظار $O(n \log n)$ خواهد بود.

۳. یافتن قله دو بعدی. یک ماتریس $n \times n$ را در نظر بگیرید. می‌خواهیم در این ماتریس درایه‌ای بیابیم که از چهار درایه‌های مجاورش کوچکتر نباشد. الگوریتمی بر اساس روش تقسیم و حل ارائه می‌کنیم به این صورت که در هر مرحله ستون میانی را انتخاب می‌کنیم. در آن ستون ماکزیمم مطلق را

برمی‌گزینیم. اگر قله بود کار تمام است. در غیر این صورت سمت افزایشی را انتخاب می‌کنیم و به این شکل ورودی را نصف می‌کنیم. نشان دهید که این الگوریتم درست کار می‌کند.

همان طور که می‌دانید زمان اجرای این الگوریتم از مرتبه $O(n \log n)$ است. حال الگوریتم فوق را به این شکل اصلاح می‌کنیم. در هر تکرار از الگوریتم اگر تعداد سطرها از تعداد ستون‌ها بیشتر بود، به جای ستون میانی، سطر میانی را به عنوان محور انتخاب می‌کنیم. آیا الگوریتم دوم بهتر کار می‌کند؟ اگر این طور است زمان اجرای الگوریتم دوم را تحلیل کنید.

۴. محاسبه ریشه. اعداد طبیعی m و n و عدد گویای $0 < u < 1$ با نمایش دودویی داده شده است. تعداد ارقام عدد u را با $|u|$ نمایش می‌دهیم. می‌خواهیم ریشه m -ام عدد u را تا n رقم از نمایش دودویی آن گزارش کنیم. بهترین الگوریتمی که می‌توانید ارائه دهید و زمان اجرای آن را تحلیل کنید.

۱. الگوریتم اقلیدس. الگوریتم یافتن ب.م.م به روش نردبانی را در نظر بگیرید. ثابت کنید این الگوریتم در زمان لگاریتمی نسبت به بزرگی اعداد یا زمان خطی نسبت به طول نمایش آن‌ها به پایان می‌رسد (راهنمایی: از دنباله فیبوناچی استفاده کنید).