باسمه‌تعالی

دوره‌ی تابستانی المپیاد کامپیوتر

آزمون الگوریتم

پنج‌شنبه ۲۹ مرداد ۱۳۹۴ | مدرسین: مهرجردی، ملکی | وقت: ۵ ساعت

توضیحات:

- آزمون شامل ۷ پرسش است. ۴ پرسش در این برگه و ۳ پرسش دیگر در برگه‌ی دوم هستند. برگه‌ی دوم پس از گذشت دو ساعت و نیم از شروع آزمون به شما داده خواهد شد.
- هر پرسش ۲۵ امتیاز دارد.
- در این آزمون تمامی عمل‌گرهای جمع، ضرب، تفریق و تقسیم از $O(1)$ انجام می‌شوند (مستقل از بزرگی اعداد).

مسأله‌ی یکم .... ۲۵ امتیاز

یک گراف ساده‌ی جهت‌دار وزن‌دار داریم. می‌خواهیم کوتاه‌ترین مسیر از s به t را پیدا کنیم. وزن مسیر بین دو رأس u,v به این صورت به دست می‌آید:

> فرض کنید یال‌های مسیر به ترتیب $e_1,e_2,\ldots,e_k$ باشد. در این صورت وزن مسیر برابر با $\sum_{i=1}^{k} e_i \times 2^i$ است.

الگوریتمی از $O(n^2)$ ارائه دهید که کم‌وزن‌ترین مسیر بین دو رأس s,t را محاسبه کند.

مسأله‌ی دوم .... ۲۵ امتیاز

درختی با k برگ داریم. فاصله‌ی دوبه‌دوی برگ‌ها به ما داده شده است. تعداد رأس‌های درون درخت را با الگوریتمی با پیچیدگی زمانی و حافظه از $O(k^2)$ بیابید.

مسأله‌ی سوم .... ۲۵ امتیاز

در یک مغازه‌ی عروسک‌فروشی، n عروسک به ترتیب با اندازه‌های $a_1,a_2,\ldots,a_n$ ، گنجایش‌های $b_1,b_2,\ldots,b_n$ و قدمت‌های $c_1,c_2,\ldots,c_n$ وجود دارد. می‌دانیم در عروسک iـام می‌توانیم حداکثر یک عروسک با اندازه‌ی کم‌تر یا مساوی b_i قرار دهیم ($b_i < a_i$). دقت کنید عروسک‌ها می‌توانند زنجیروار در یک‌دیگر قرار بگیرند. برای مثال عروسک ۱ در عروسک ۲ و عروسک ۲ در عروسک ۳ قرار بگیرد. اگر عروسکی را درون عروسک دیگری قرار بدهیم، برای خارج کردن عروسک درونی هزینه‌ای نمی‌پردازیم؛ زیرا مغازه‌دار متوجه نمی‌شود. برای مثال اگر عروسک ۱ درون عروسک ۲ و عروسک ۲ درون عروسک ۳ باشد، کافی است فقط هزینه‌ی عروسک ۳ را بپردازیم.

قیمت هر عروسک یک تومان است و ما تنها دو تومان پول داریم. الگوریتمی از $O(n^2)$ ارائه دهید؛ طوری که مجموع قدمت عروسک‌هایی که از مغازه خارج کرده‌ایم بیشینه شود (در صورتی که الگوریتمی از $O(n^3)$ ارائه دهید، می‌توانید تا حداکثر ۱۵ نمره بگیرید).

مسأله‌ی چهارم . . . . ۲۵ امتیاز

یک گراف جهت‌دار وزن‌دار G با مجموعه‌ی رئوس $V(G)$ و مجموعه‌ی یال‌های $E(G)$ داریم. یال جهت‌دار از u به v را با (u,v) نشان می‌دهیم. وزن یال از u به v را $w(u,v)$ تعریف می‌کنیم. به یک زیرگراف H از گراف G، MST_r گوییم؛ هرگاه گراف زمینه‌ی H درخت باشد و در H از r به تمام رئوس G مسیر جهت‌دار وجود داشته باشد و مجموع وزن یال‌های H کمینه باشد. $f(G,r)$ برابر با مجموع وزن یال‌های MST_r در گراف G است. می‌خواهیم $f(G,r)$ را پیدا کنیم. "ادموند" الگوریتم بازگشتی زیر را برای این کار ارائه می‌دهد:

> برای هر رأس به جز r مانند v، یال ورودی با وزن کمینه به v را در نظر بگیرید (اگر چند یال با وزن کمینه وجود داشت، یکی را به دل‌خواه انتخاب می‌کنیم) و فرض کنید آن یال از رأس $\pi(v)$ خارج می‌شود. به زیرگراف با مجموعه رئوس $V(G)$ و مجموعه‌ی یال‌های مذکور، H' می‌گوییم. اگر در H' دور جهت‌دار وجود نداشت، H' پاسخ مسئله است؛ در غیر این صورت یکی از دورهای H' را به دل‌خواه در نظر گرفته و آن را C بنامید. گراف G' را به روش زیر می‌سازیم:
>
> > $V(G')$ را برابر $\{V(G)-V(C)\}\cup a$ در نظر می‌گیریم که a یک رأس جدید است. یال‌های G' را نیز به این صورت می‌سازیم:
> >
> > - اگر (u,v) یک یال در $E(G)$ بود؛ طوری که $u\notin C$ و $v\in C$، یک یال در $E(G')$ از u به a با وزن $w(u,v)-w(\pi(v),v)$ می‌گذاریم.
> > - اگر (u,v) یک یال در $E(G)$ بود؛ طوری که $u\in C$ و $v\notin C$، یک یال در $E(G')$ از a به v با وزن $w(u,v)$ می‌گذاریم.
> > - اگر (u,v) یک یال در $E(G)$ بود؛ طوری که $u\notin C$ و $v\notin C$، یک یال در $E(G')$ از u به v با وزن $w(u,v)$ می‌گذاریم.
>
> حال الگوریتم را روی G' اجرا می‌کنیم تا $f(G',r)$ به دست آید. مجموع وزن یال‌های C را $g(C)$ در نظر می‌گیریم. در انتها مقدار $f(G,r)$ را $f(G',r)+g(C)$ اعلام می‌کنیم.

ثابت کنید الگوریتم ادموند درست کار می‌کند.