

ساختمان داده‌ها و الگوریتم‌ها

تمرین پنجم - گراف

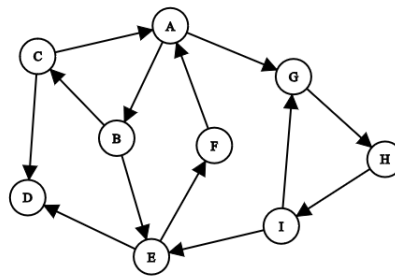
ماردین نیچی، شایان کاشفی

تاریخ تحویل: ۱۴۰۲/۱۰/۱۲

۱۵ نمره

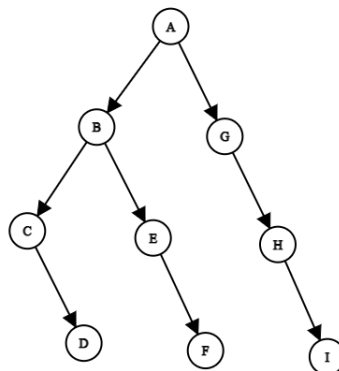
۱. سوال ساده

پیمایش‌های BFS و DFS را با شروع از راس A بر روی گراف زیر اجرا کنید. ترتیب پیمایش راس‌ها را مشخص کنید و درخت به‌دست آمده از هر الگوریتم را بکشید؛ همچنین در هر دو الگوریتم، cross-edge و back-edge ها را مشخص کنید. (در صورتی که در یک مرحله می‌توانستید بیش از یک گره را انتخاب کنید، به ترتیب حروف الفبا، گره انتخاب شده را مشخص کنید)



پاسخ:

DFS :

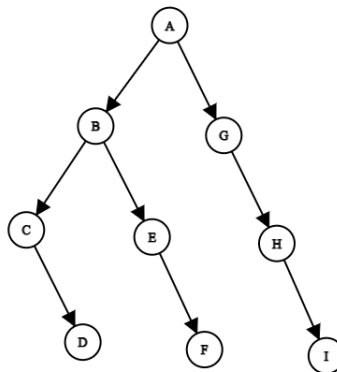


CrossEdges : $E \rightarrow D, I \rightarrow E$

BackEdges : $C \rightarrow A, F \rightarrow A, I \rightarrow G$

Order : $A, B, C, D, E, F, G, H, I$

BFS :



CrossEdges : $E \rightarrow D, I \rightarrow E$

BackEdges : $C \rightarrow A, F \rightarrow A, I \rightarrow G$

Order : $A, B, G, C, E, H, D, F, I$

۱۵ نمره

۲. جاده‌های حیاتی

اصغر در کشوری با n شهر و m جاده دو طرفه بین شهرها قرار دارد. او در حال سفر کردن بین شهرها است که متوجه می‌شود وزارت راه و شهرسازی مشغول تعمیر یکی از جاده‌هاست. اصغر نمی‌تواند از جاده در حال تعمیر عبور کند و به همین دلیل امکان دارد سفرش خراب شود. سفر اصغر در صورتی خراب می‌شود که دو شهر مانند u و v وجود داشته باشند که که اصغر نتواند از یکی به دیگری برود. لیست جاده‌هایی که اگر در حال تعمیر باشند سفر اصغر خراب می‌شود را بیابید.

پاسخ:

مسئله تعداد پال‌های برشی گراف متناظر با کشور را می‌خواهد. برای پیدا کردن پال برشی، ابتدا بر روی گراف الگوریتم DFS را اجرا می‌کنیم. در هنگام اجرا شدن الگوریتم DFS به ازای هر راس i یک مقدار $dp[i]$ نگه می‌داریم که نشان دهنده ی این است که در زیر درخت راس i ، بالا ترین backedge به چه ارتفاعی وصل می‌شود. مقدار $dp[i]$ برای هر راس این گونه حساب می‌شود که مقدار یک راس برابر خواهد شد با ماکسیمم مقداری که فرزندان آن راس دارند و پال‌هایی که از خود آن راس به پدرانش وصل است. حال یک پال که بین راس v و $parent[v]$ برشی است اگر و فقط اگر $dp[v]$ مقدارش بالا تر از ارتفاع v نباشد. یعنی پالی در زیر درخت وجود نداشته باشد که به راسی بالا تر از v وصل باشد. زمان اجرای این الگوریتم $O(n + m)$ است.

۱۵ نمره

۳. شجره‌نامه

علی به تازگی شجره‌نامه خانوادگی‌شان را پیدا کرده است و با جدیت تمام مشغول بررسی کردن روابط خانوادگی است. (شجره‌نامه را می‌توان مانند یک درخت n راسی ریشه‌دار فرض کرد). پدر بزرگ علی وقتی او را در این وضعیت می‌بیند، برایش سوال می‌شود که نسل جدید چقدر از خاندان خود باخبر هستند. به همین دلیل تصمیم می‌گیرد از علی q سوال بپرسد. هر سوال او به این شکل است که اسم دو نفر از کل خاندان را به علی می‌دهد و علی باید مشخص کند که آیا یکی از این دو نفر جد دیگری هست یا خیر. پدر بزرگ پیر شده‌است و حوصله ندارد منتظر بماند، به همین دلیل علی باید کل سوال‌ها را در $O(n + q)$ حل کند به علی کمک کنید تا پدر بزرگ خود را خوشحال کند. (n تعداد کل افراد خاندان است)

پاسخ:

پدربزرگ دو راس در یک درخت به ما میدهد و از ما میخواهد تشخیص دهیم یکی جد دیگری است یا خیر. برای حل این مسئله از ریشه شروع میکنیم و روی درخت DFS میزنیم. برای هر راس $startingTime$ و $finishingTime$ آن را بترتیب در آرایه های s و f ذخیره میکنیم. راس u جد راس v است اگر $s[u] \leq s[v]$ و $f[u] \geq f[v]$ باشد. اگر این دو شرط برقرار باشند، نشان میدهد که v را قبل از u دیده‌ایم ولی قبل از خارج شدن از u در زیردرخت u قرار دارد پس u جد v است. پس برای حل مسئله کافی است در ابتدا با یک DFS مقادیر $startingTime$ و $finishingTime$ را برای همه ی راس ها حساب کنیم ($O(n)$). و برای هر پرسش دو شرط بالا را چک میکنیم تا ببینیم یکی در زیردرخت دیگری قرار دارد یا خیر ($O(q)$)

۴. گشت زنی

۱۵ نمره

یک جدول $n \times n$ به شما داده می‌شود که ارتفاع خانه (i, j) آن برابر $h_{i,j}$ است. دو خانه را مجاور می‌گوییم اگر حداقل یک راس مشترک داشته باشند. یک بلوک در جدول که مجموعه‌ای از خانه‌ها مثل S است، به این شکل تعریف می‌شود:

- همه خانه‌های S هم‌ارتفاع باشند.
- مجموعه خانه S ، همبند باشند (یعنی از هر خانه در S ، مسیری درون S به هر خانه دیگری از این مجموعه وجود داشته باشد).
- اگر $s \in S$ و $t \notin S$ و t مجاور باشند، در این صورت $h_t \neq h_s$.

به یک بلوک قله می‌گوییم هرگاه برای هر خانه مجاور بلوک، ارتفاع آن خانه از ارتفاع خانه‌های بلوک کمتر باشد.
به یک بلوک دره می‌گوییم هرگاه برای هر خانه مجاور بلوک، ارتفاع آن خانه از ارتفاع خانه‌های بلوک بیشتر باشد.
تعداد قلّه‌ها و درّه‌ها در جدول را پیدا کنید.

پاسخ:

برای این سؤال فرض کنید گرافی می‌سازیم که هر رأس آن متناظر یک خانه از جدول باشد. بین دو رأس یالی می‌گذاریم اگر و تنها اگر عدد نوشته‌شده در خانه‌های متناظرشان برابر باشد و همچنین این دو خانه مجاور رأسی باشند. بلوک در این مسئله معادل مؤلفه‌ی همبندی در چنین گرافی است. این گراف $n \times n$ تا رأس دارد و تعداد یالهایش حداکثر برابر تعداد جفت خانه‌های مجاور است اگر از روش DFS برای به‌دست آوردن مؤلفه‌ها استفاده کنیم می‌توانیم سپس به ازای هر مؤلفه با چک کردن خانه‌های مرزی‌اش تعیین کنیم قله است و یا دره و در $O(n^2)$ الگوریتمی برای این کار ارائه داده‌ایم.

۵. هزارتو

۲۰ نمره

اکبر در یک هزارتو به شکل جدول $n \times n$ گم شده است. هر خانه این هزارتو یا خالی است و می‌توان به داخل آن خانه رفت و یا مسدود شده و نمی‌توان وارد آن شد. اکبر در خانه (x, y) قرار دارد و خروج در خانه (X, Y) قرار دارد. اکبر می‌خواهد خودش را به خانه خروج برساند ولی در این هزارتو تنها نیست و K زامبی در خانه های x_i, y_i قرار دارند که به دنبال او هستند تا او را هم به زامبی تبدیل کنند. اکبر و زامبی‌ها در هر لحظه یا می‌توانند از یک خانه در هزارتو به یک خانه مجاور ضلعی بروند یا در خانه فعلی خود بمانند. الگوریتمی بیابید که در مرتبه زمانی $O(n^2)$ مشخص کند آیا اکبر می‌تواند به خانه خروج برسد؟ (فرض کنید زامبی‌ها در هر لحظه بهترین تصمیم را می‌گیرند. توجه کنید که اگر اکبر و یکی از زامبی‌ها باهم به خانه خروج برسند، زامبی همچنان اکبر را می‌گیرد و به زامبی تبدیل می‌کند)

پاسخ:

اگر فاصله اکبر از خانه خروج، از مینیمم فاصله‌ی همه‌ی زامبی‌ها تا خانه خروج کوچکتر باشد، موفق میشود فرار کند اما در غیر این صورت زامبی‌ها او را می‌گیرند. در حالت اول اگر اکبر کوتاهترین مسیر را انتخاب کند، در هر مرحله یک واحد به خروج نزدیکتر میشود، پس حتما میتواند فاصله‌ی حداقل یک واحدی با زامبی‌ها حفظ کند. در حالت دوم زامبی با مینیمم فاصله از خانه خروج میتواند به خانه خروج برود و در آنجا بماند و راه خروج را مسدود کند. حال برای پیدا کردن فاصله زامبی‌ها و اکبر از خانه خروج، هزارتو را به یک گراف مدل میکنیم که هر خانه یک راس باشد و برای هر دو خانه مجاور خالی یک یال بین راس‌های متناظر آن‌ها در گراف رسم میکنیم. اکنون با انجام BFS از راس متناظر با خانه خروج میتوان فاصله تمام راس‌ها را از این راس حساب کرد. پس مساله در مرتبه زمانی $O(n + m)$ حل شد.

۶. سفر در اقیانوس

۲۰ نمره

کیوان جهت پیدا کردن گنجی بزرگ به اقیانوسی سفر کرده است. این اقیانوس n جزیره دارد و جزیره i ام در مختصات (x_i, y_i) قرار دارد. برای یافتن گنج لازم است کیوان به تمامی جزیره‌ها برود. کیوان همراهش یک قایق دارد و هنگامی که در جزیره‌ای قرار دارد، یکی از چهار جهت (بالا، پایین، چپ و راست) را انتخاب می‌کند و قایقش در همان جهت به حرکت ادامه می‌دهد تا به جزیره‌ای دیگر برسد (به این فرایند مسافرت می‌گوییم).

دقت کنید او نمی‌تواند در میانه راه توقف کند. ضمناً کیوان قادر است بین برخی از جفت جزیره‌ها مثل (a, b) یک خط ارتباطی پیشرفته ایجاد که به کمک آن می‌تواند هنگامی که در a قرار دارد به همراه قایقش به جزیره b برود و همین‌طور بالعکس (یعنی از b به a). کیوان می‌خواهد کمترین تعداد مسیرهای ارتباطی ویژه را ایجاد کند که به کمک آن‌ها بتواند از هر جزیره‌ای، با تعدادی مسافرت بین جزیره‌ها به هر جزیره دیگری برسد. به او در پیدا کردن کمترین تعداد مسیرهای ارتباطی ویژه مورد نیاز کمک کنید.

پاسخ:

مسئله را به یک گراف مدل می‌کنیم. هر جزیره را یک راس در نظر می‌گیریم و بین دو راس u و v یال رسم می‌کنیم اگر و تنها اگر یکی از دو حالت $i_u = i_v$ یا $j_u = j_v$ برقرار باشد. به عبارتی دیگر بین دو راس یال رسم می‌کنیم اگر و تنها اگر بتوان با یک بار مسافرت کردن از یکی به دیگری رسید. بدیهی است که با قرار داشتن در یک راس (جزیره) و با حرکت روی یال‌ها (مسافرت) به بقیه راس‌هایی که در آن مولفه همبندی هستند می‌توان رسید. حال برای این که به همه ی راس‌ها دسترسی داشته باشند باید بین مولفه‌های همبندی یال (خط ارتباطی ویژه) رسم کنیم بطوری که با کمترین تعداد رسم یال، گراف همبند شود. از ریاضیات گسسته میدانیم که برای همبند کردن یک گراف با k مولفه ی همبندی حداقل به $k - 1$ یال نیاز داریم. پس مساله ما به یافتن تعداد مولفه‌های همبندی در یک گراف ساده تبدیل میشود. برای پیدا کردن تعداد مولفه‌های همبندی می‌توانیم هم از BFS و هم از DFS استفاده کنیم. در واقع باید دنباله‌ای از DFS‌ها را روی گراف اجرا کرد: اولین DFS را روی یکی از رئوس گراف مانند v اجرا می‌کنیم و با این کار تمام رئوس متعلق به اولین مولفه همبندی پیدا خواهند شد. سپس اولین راس از رئوس باقی مانده که دیده نشد $mark[u] = false$ پیدا می‌کنیم و روی آن هم DFS می‌زنیم. بدین ترتیب رئوس متعلق به دومین مولفه همبندی هم پیدا می‌شود و همین کار را ادامه می‌دهیم تا جایی که هیچ راسی که دیده نشده است، باقی نماند. اردر این الگوریتم از $O(n + m)$ است. در این مدل سازی خاص که ما از مسئله داشتیم تعداد یال‌ها از $O(n^2)$ بود پس اردر نهایی از $O(n^2)$ است. برای تمرین بیشتر می‌توانید مدل‌سازی ارائه دهید که تعداد یال‌ها از $O(n)$ باشد.