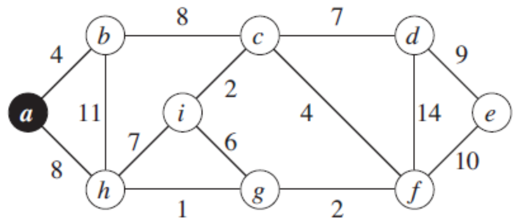
جواب سوالات آزمون میانترم

درس طراحی الگوریتم – دانشکده ریاضی	
دانشگاه صنعتی خواجه نصیرالدین طوسی. پاییز ۱۴۰۲	زمان امتحان: ۸۰ دقیقه

۱. توابع زیر را بر اساس رشد مجانبی از کوچک به بزرگ مرتب کنید.

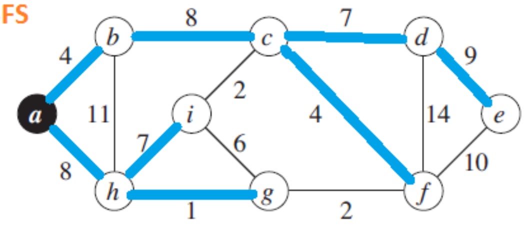
- $b(n) = n^{1/\log n} = 2$
- $f(n) = (\sqrt{2})^{\log n} = n^{\log \sqrt{2}} = \sqrt{n}$
- $g(n) = (\log n)! < 2^{\log^2 n}$
- $t(n) = 2^{\sqrt{2n}}$
- $a(n) = (1.5)^n$

۲. با توجه به گراف زیر به سوالات پاسخ دهید.

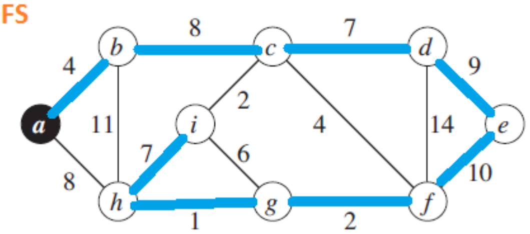


- درخت حاصل از پیمایشهای DFS و BFS و درخت کوتاهترین مسیر با شروع از راس a را رسم کنید (نیازی به نوشتن مراحل میانی نیست.) برای اولویت از ترتیب الفبا استفاده کنید.

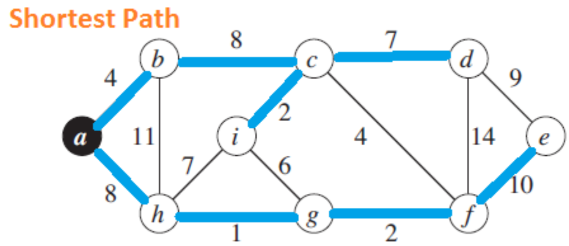
BFS

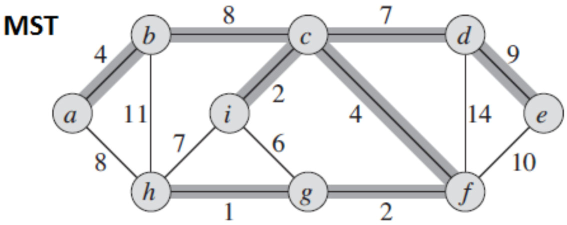


DFS



- درخت فراگیر کمینه را برای گراف بالا بدست آورید.





۳. گراف جهت دار $G = (V, E)$ قویا همبند است اگر و فقط اگر از هر راس $u \in V$ به هر راس $v \in V$ مسیری وجود داشته باشد. نشان دهید که در زمان $O(|V| + |E|)$ می‌توان چک کرد که آیا G قویا همبند است یا نه. فرض کنید گراف بصورت لیست مجاورتی داده شده است.

با توجه به تعریف داده شده برای قویا همبند بودن، یک تعریف معادل این است که گراف جهت دار G قویا همبند است اگر و فقط اگر برای هر راس u داشته باشیم، u به همه رئوس دیگر مسیر دارد و از همه رئوس دیگر هم به u مسیر وجود دارد. بر اساس این تعریف معادل، یک الگوریتم پیشنهاد می‌دهیم.

- راس دلخواه $u \in V$ را انتخاب می‌کنیم و الگوریتم BFS را با شروع از u اجرا می‌کنیم.
- در قدم دوم، جهت یالهای گراف G را عوض می‌کنیم و دوباره BFS را با شروع از راس u فراخوانی می‌کنیم.
- اگر در هر دو اجرای BFS همه رئوس گراف ملاقات شد گزارش می‌کنیم که گراف ورودی قویا همبند است در غیر اینصورت گزارش می‌کنیم که گراف ورودی قویا همبند نیست.

لازم به ذکر است که در الگوریتم بالا از DFS هم می‌توان استفاده کرد. چون فقط دو بار BFS را فراخوانی کردیم و زمان اجرای BFS حداکثر $O(|E| + |V|)$ است و از طرف دیگر عوض کردن جهت یالها در زمان $O(|V|)$ قابل انجام است پس الگوریتم پیشنهادی زمان اجرایش $O(|E| + |V|)$ است.

۴. یک تطابق در گراف $G = (V, E)$ به مجموعه ای از یالها گفته می شود که هیچ راس مشترک نداشته باشند. یک تطابق بیشینه در G بیشترین تعداد یال را در میان همه تطابقها دارد. الگوریتم حریصانه زیر را در نظر بگیرید. مجموعه F در ابتدا تهی است. یال دلخواه (u, v) را انتخاب کن و در F قرار بده. سپس u و v و یالهای روی آن را از گراف حذف کن. برای گراف که باقیمانده همین کار را تکرار کن. یعنی یک یال دلخواه از آن انتخاب کن و در F قرار بده. این پروسه را تکرار کن تا زمان که گراف تهی شود. نشان دهید در انتهای کار $|F| \geq \frac{1}{2}\nu(G)$ وقتی که $\nu(G)$ اندازه تطابق بیشینه در گراف است.

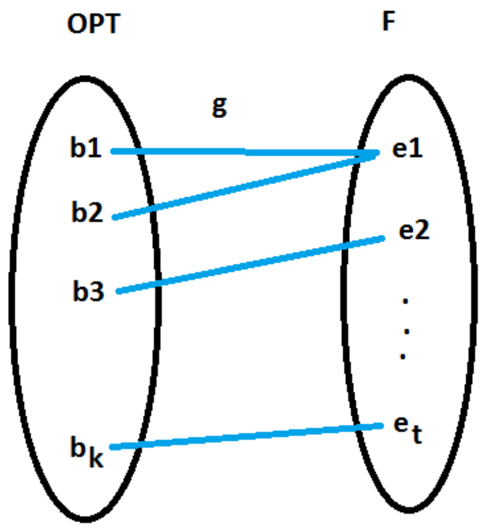
فرض کنید $OPT = \{b_1, b_2, \cdots, b_k\}$ یک تطابق بیشینه باشد و $F = \{e_1, \cdots, e_t\}$ خروجی الگوریتم حریصانه باشد. بنا به تعریف داریم

$$\nu(G) = |OPT|$$

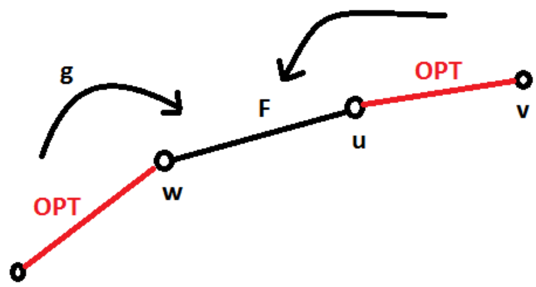
نشان می‌دهیم تناظر

$$g : OPT \to F$$

وجود دارد بطوریکه برای هر یال $e \in F$ حداکثر 2 یال x و y در OPT وجود دارد بطوریکه $g(x) = g(y) = e$. این نشان می‌دهد که $|F| \geq \frac{1}{2}|OPT| = \frac{1}{2}\nu(G)$. شکل زیر را ببینید.



اگر داشته باشیم $e \in F \cap OPT$ آنگاه قرار می‌دهیم $g(e) = e$. فرض کنید یال $b = (u, v)$ که عضوی از OPT است در F وجود ندارد. یال b توسط الگوریتم حریصانه انتخاب نشده چون حتما w وجود دارد بطوریکه $e = (u, w) \in F$. در این حالت قرار می‌دهیم $g(b) = e$. اگر اینطور نباشد حتما z وجود دارد بطوریکه $e' = (v, z) \in F$. در این حالت قرار می‌دهیم $g(b) = e'$. دقت کنید ممکن است هر دو حالت رخ داده باشد. اگر هیچکدام از این دو حالت رخ ندهد، الگوریتم حریصانه می‌بایست یال b را به جمع F اضافه می‌کرد ولی نکرده است که متناقض است. دقت کنید تناظر g نمی‌تواند بیشتر از دو یال را به $e \in F$ نسبت دهد. این ادعای ما را ثابت می‌کند. به شکل زیر توجه کنید.



موفق باشید