به نام خدا


دانشگاه تهران، دانشکده مهندسی برق و کامپیوتر
تحلیل و طراحی الگوریتم‌ها

راه حل تمرین کتبی چهارم

---

۱. (آ)

- مرحله اول :

$S = \{\}$
$d[1] = \infty$
$d[2] = \infty$
$d[3] = \infty$
$d[4] = \infty$
$d[5] = \infty$
$d[6] = \infty$

- مرحله دوم :

$S = \{1\}$
$d[1] = 0$
$d[2] = 7$
$d[3] = 9$
$d[4] = \infty$
$d[5] = \infty$
$d[6] = 14$

- مرحله سوم :

$S = \{1, 2\}$
$d[1] = 0$
$d[2] = 7$
$d[3] = 9$
$d[4] = 22$
$d[5] = \infty$
$d[6] = 14$

- مرحله چهارم :

$S = \{1, 2, 3\}$
$d[1] = 0$
$d[2] = 7$
$d[3] = 9$
$d[4] = 20$
$d[5] = \infty$
$d[6] = 11$

- مرحله پنجم :

$S = \{1, 2, 3, 6\}$
$d[1] = 0$
$d[2] = 7$
$d[3] = 9$
$d[4] = 20$
$d[5] = 20$
$d[6] = 11$

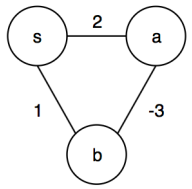
- مرحله ششم :

$S = \{1, 2, 3, 6, 4\}$
$d[1] = 0$
$d[2] = 7$
$d[3] = 9$
$d[4] = 20$
$d[5] = 20$
$d[6] = 11$

- مرحله هفتم :

$S = \{1, 2, 3, 6, 4, 5\}$
$d[1] = 0$
$d[2] = 7$
$d[3] = 9$
$d[4] = 20$
$d[5] = 20$
$d[6] = 11$

(ب) در گراف زیر برای به دست آوردن کوتاه ترین فاصله از راس s به دو راس دیگر از دایکسترا استفاده می کنیم. نتیجه ی این الگوریتم به این صورت خواهد بود :



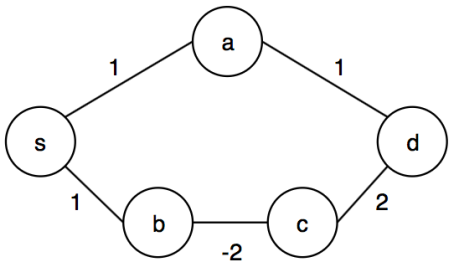
$d[a] = 2$ ،$d[b] = 1$

که این غلط است و جواب درست به این صورت است :

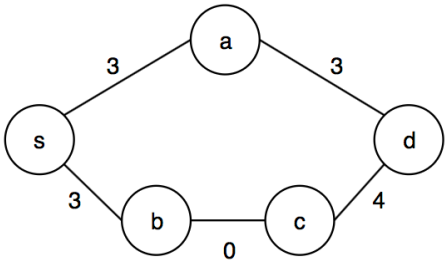
$d[a] = -2$ ،$d[b] = -1$

علت غلط عمل کردن الگوریتم در اینجا آن است که در فرایند اجرای دایکسترا در مرحله ی نهایی کردن فاصله تا یک راس اطمینان داریم افزودن هر یال دیگری به مسیر به علت مثبت بودن وزن یال باعث افزایش مسیر می شود پس می توانیم فاصله را قطعی کنیم. اما اینجا یال منفی می تواند از فاصله ی به دست آمده بکاهد.

(ج) گراف زیر را در نظر می گیریم.



تمام وزن هارا دو واحد افزایش می دهیم و الگوریتم دایکسترا را روی آن اجرا می کنیم.



همانطور که می بینیم در گراف اصلی کوتاه ترین مسیر از s به d مسیر sbcd است اما در گراف جدیدی که ساختیم کوتاه ترین مسیر مسیر sad است که این نتیجه غلط است.و به آن علت است که عدد افزوده شده به طول مسیر با تعداد یال هایی که در مسیر وجود دارند رابطه دارد.

۲. گراف'Gرا به این شکل می سازیم : هر کدام از مولفه های همبندیFرا یک راس در نظر میگیریم و رئوسی که در هیچ مولفه هبندی نیستند به همان شکل می گذاریم. یال های بین رئوس عادی را به همان شکل میگذاریم و یال بین یک راس عادی و یک راس نماینده مولفه همبندی را یال با کمترین وزن بین آن راس و رئوس آن مولفه همبندی در نظر میگیریم. اگر بین رئوس یک مولفه همبندی با رئوس یک مولفه همبندی دیگر یال وجود داشته باشد، یال با کمترین وزن را به عنوان یال بین۲راس آن دو مولفه همبندی در گراف'Gدر نظر میگیریم. حال کافیست MST را در گراف'Gپیدا کنیم. برای پیاده سازی می توان ابتدا وزن یال های F را صفر کرد سپس از کروسکال یا پرایم استفاده کرد.

۳. یک گراف می سازیم که دانشجویان کلاس راس های آن است و بین هر دو نفری که باهم تقلب کرده اند یک یال قرار می دهیم. در صورتی که گراف به دست آمده همبند نبود ابتدا با DFS مولفه های همبندی را پیدا می کنیم و سپس الگوریتم زیر را روی هر مولفه اجرا می کنیم:
با شروع از یک راس دلخواه الگوریتم BFS را اجرا می کنیم. فاصله ی هر راس تا راس اولیه برابر است با تعداد یال هایی که بینشان وجود دارد. در هر مرحله اگر بین دو راس با فاصله ی یکسان از مبدا یالی وجود داشته باشد نمی توانیم افراد را به دو گروه تقسیم کنیم و اگرنه راس هایی با فاصله ی زوج از مبدا را در یک کلاس و راس هایی با فاصله ی فرد از مبدا را در گروهی دیگر قرار می دهیم.

۴. از یک راس دلخواهBFSمی زنیم و دورترین راس به آن راس را پیدا می کنیم .(v) راسvقطعا یک سر یک بلندترین مسیر درون درخت خواهد بود. حال ازvیکBFSدیگر میزنیم و دورترین راس را بهvپیدا میکنیم(u). مسیر بینuوvبلند تریم مسیر در درخت خواهد بود.

۵. یک گراف جهتدار وزندار تشکیل می‌دهیم که راس ها شهرها هستند و یال‌ها جاده‌ها و وزن یال بین شهر i و j برابرS(i،j)می‌باشد. حال وزن جاده‌های که در آن ها صرافی وجود دارد را منهای ۲ می‌کنیم و وزن آن جاده هایی که به شهرهایی که در آن‌ها بانک وجود دارد میرسند را نیز منهای ۱ می‌کنیم. حال روی گراف به وجود آمده با استفاده از الگوریتم Bellman Ford تشخیص می‌دهیم که آیا دوری با طول منفی وجود دارد یا خیر و اگر وجود داشته باشد به این معناست که علی می‌تواند با شروع از نمک‌آباد و چرخیدن در شهرها، با پول بیشتری به نمک‌آباد بازگردد.

۶. برای هر یال گراف، یک یال با جهت برعکس متناظر با آن رسم می‌کنیم و وزن یال اصلی را۰ و وزن یال اضافه شده را برابر ۱ قرار می‌دهیم. حال الگوریتم دایکسترا را با شروع از source برای گراف حاصل اجرا می‌کنیم. که در نهایت هزینه‌ی رسیدن به target را به ما می‌دهد و این برابر همان تعداد یال هایی است که باید reverse شوند تا از source به target مسیر داشته باشیم.