



ساختمان داده‌ها و الگوریتم‌ها

نیم‌سال اول ۱۴۰۰-۱۴۰۱
مدرس: مسعود صدیقین

محمدجواد ماهرالنقش

کوتاه‌ترین مسیر در گراف وزن‌دار

یادآوری جلسه بیست‌و‌چهارم

در جلسه گذشته، در مورد مساله پیدا کردن کوتاه‌ترین مسیر در گراف وزن‌دار صحبت کردیم و الگوریتم دایسترا را مورد بررسی قرار دادیم. مساله‌ای که به آن پرداختیم به این‌صورت بود که گراف وزن‌دار و جهت‌دار G داده شده است، هدف پیدا کردن کوتاه‌ترین مسیر بین راس v_1 و راس v_n است.

سختی پیدا کردن فاصله راس ۱ تا راس n معادل سختی پیدا کردن فاصله راس ۱ تا همه راس‌های دیگر است. برای حل این مساله الگوریتم دایسترا را معرفی کردیم:

الگوریتم دایسترا: در ابتدا راس ۱ را به مجموعه S اضافه می‌کنیم. آرایه $dist$ را به‌گونه‌ای در نظر می‌گیریم که $dist[i]$ نشان‌دهنده فاصله راس i تا راس ۱ است. در ابتدا $dist$ راس‌هایی که از راس ۱ به آن‌ها یال وجود دارد را برابر وزن آن یال قرار داده و $dist$ مربوط به سایر راس‌ها را ∞ می‌گذاریم.

در هر مرحله، در بین تمامی راس‌های عضو $V - S$ ، راس با $dist$ کمینه را انتخاب و به مجموعه S اضافه می‌کنیم. فاصله آن راس تا راس ۱ برابر $dist$ مربوط به آن در این لحظه می‌باشد. سپس $dist$ راس‌های خارج از S که این راس به آن‌ها یال دارد را به روز می‌کنیم. قطعه کد زیر چگونگی الگوریتم دایسترا را نشان می‌دهد:

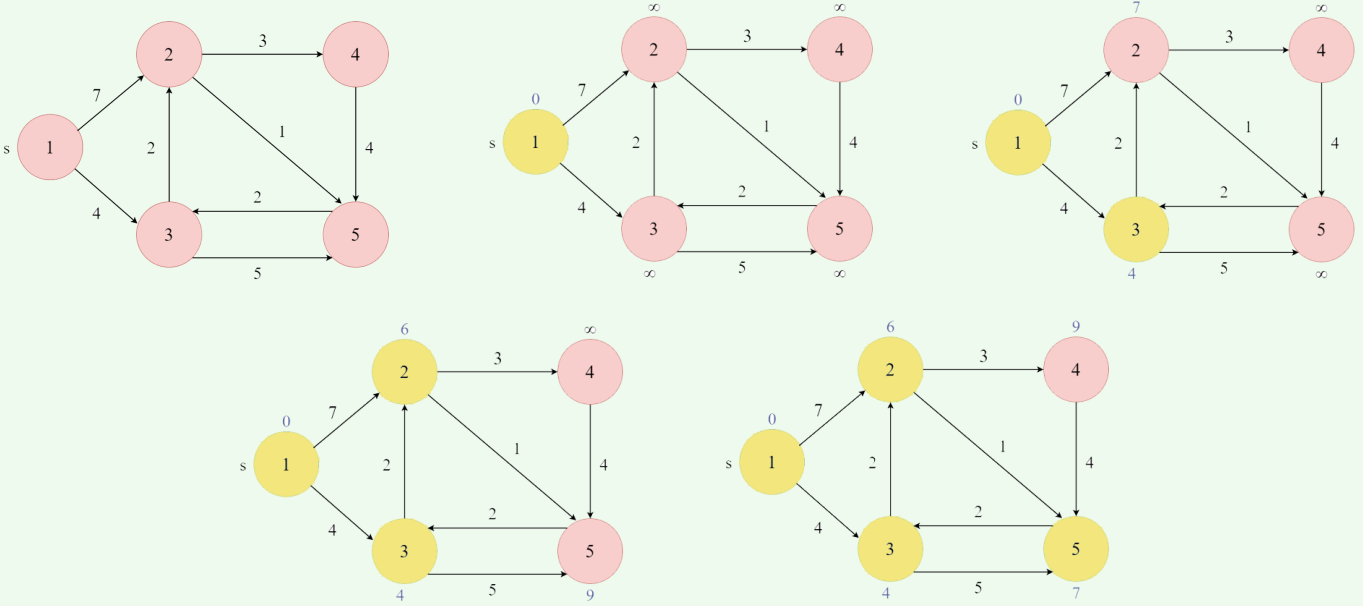
```
S = { ۱ }

forall j in Neighbors[۱] : dist[j] = weight(۱, j)

forall j not in Neighbors[۱] : dist[j] = ∞

for (i : ۲ → n) {
    . let k be a vertex (راس) in V/S with minimum dist (S نیستند)
    . add k to S
    . for every j in Neighbors(k) {
        . if (weight(k, j) + dist[k] < dist[j]) {
        .     dist[j] = dist[k] + weight(k, j)
        . }
    . }
```

مثال: می‌خواهیم کوتاه‌ترین مسیر از راس ۱ به بقیه راس‌ها را پیدا کنیم.



حال مرتبه زمانی الگوریتم را بررسی می‌کنیم. اگر برای به دست آوردن مقدار کمینه $dist$ از پیمایش روی آرایه و به دست آوردن کمینه استفاده کنیم، در این صورت هر عملیات پیدا کردن کمینه از $O(n)$ می‌باشد. پس الگوریتم دایسترا در پیاده‌سازی با استفاده از ماتریس مجاورت و لیست مجاورت از مرتبه $O(n^2)$ خواهد بود. در پیاده‌سازی با استفاده از لیست مجاورت، اگر از هرم کمینه استفاده کنیم، هر عملیات درج، حذف کمینه و کاهش مقدار $dist$ از $O(\log n)$ است پس الگوریتم از مرتبه $O(m \log n + n \log n)$ بوده که می‌توان آن را از مرتبه $O(m \log n)$ در نظر گرفت. در این پیاده‌سازی اگر از هرم فیبوناچی بهره گیریم، هر عملیات درج و کاهش مقدار $dist$ ، از $O(1)$ و هر عملیات حذف کمینه از $O(\log n)$ است، پس الگوریتم از مرتبه $O(n \log n + m)$ خواهد بود.

پرسش: الگوریتم دایسترا مربوط به راس ۲ را برای گراف شکل زیر اجرا کنید و ترتیب اضافه شدن راس‌ها به مجموعه S و فاصله راس ۲ تا هر راس را به‌دست آورید. پاسخ‌های خود را به [این لینک](#) ارسال کنید.

