

پروژه چهارم

طراحی الگوریتم دکتر احمدی

گرگ و گوسفند

علیرضا فرزانه

این مسئله را با کاهش به مسئله max flow می توان حل کرد. اگر ظرفیت هر یال را ۱ در نظر بگیریم این مسئله همان مسئله max flow خواهد بود، زیرا وقتی flow از یالی خارج می شود در واقع ظرفیت آن یال پر شده و قابل استفاده نیست. پس در نهایت بیشترین تعداد مسیرهای ممکن edge-disjoint برابر max flow خواهد بود.

همچنین مسئله max flow به مسئله min cut به این صورت مربوط است که، حد بالای min cut ای که راس s و t را جدا می کند برابر با max flow گراف داده شده است. در نهایت پس از اجرای الگوریتم فورد فولکرسون، با استفاده از اطلاعات به دست آمده، می توان مسیرهای ممکن را استخراج کرد.

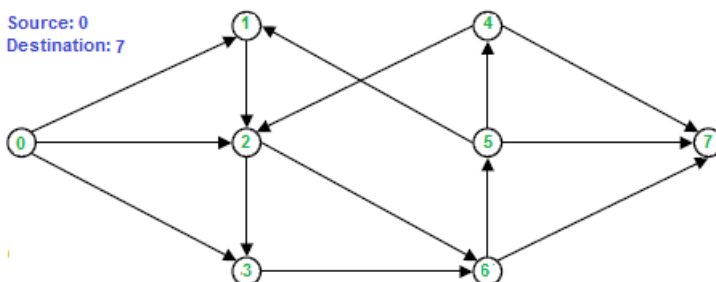
در نهایت هم اولین نتیجه به دست آمده با استفاده از networkx به صورت گرافیکی نمایش داده می شود.

پیچیدگی زمانی BFS به کار رفته برای پیدا کردن مسیر از $O(V^2)$ (از ماتریس مجاورت استفاده شده) است.

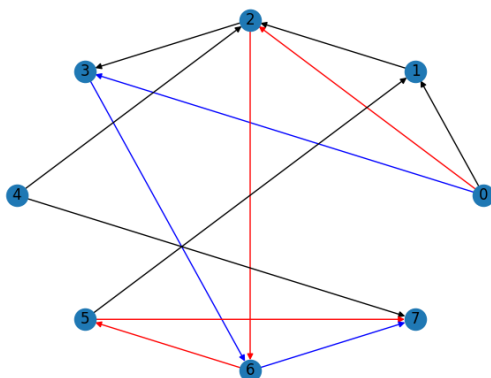
الگوریتم در کل به اندازه $O(\max \text{ flow})$ تکرار می شود.

پس پیچیدگی زمانی کلی الگوریتم از $O(\max \text{ flow} * V^2)$ خواهد بود.

نمونه ورودی:



نمونه خروجی:



در این مثال محاسبه می شود که $\max \text{ flow} = 2$ است پس در هر

لحظه دو مسیر وجود دارد که در آنها یال مشترک وجود ندارد.

max flow going from 0 to 7 is 2

possible paths:

[0, 2, 6, 5, 7] [0, 3, 6, 7]

[0, 2, 6, 5, 7] [0, 1, 2, 3, 6, 7]

[0, 1, 2, 6, 5, 7] [0, 3, 6, 7]

[0, 1, 2, 6, 5, 7] [0, 2, 3, 6, 7]

[0, 3, 6, 5, 7] [0, 2, 6, 7]

[0, 3, 6, 5, 7] [0, 1, 2, 6, 7]