مسئله : کوتاهترین مسیر بین دو راس دلخواه در یک گراف را بدست آورید.

فرض کنیم ai یال i ام و Ai مجموعه ای شامل یالهای متصل به ai است.

فرض کنیم راس x راس شروع و راس y راس پایانی باشد.

برای هر مسیر ممکن بین x و y یک n تایی می نویسیم، عضو اول این n تایی یالی است که به x متصل ، عضو دوم یالی است که به یال اول متصل است و ... و عضو آخر آن به y متصل است.

مثال: T=(t1,t2,t3,…tn)

t1 یالی است که به x متصل است.

t2 یالی است که به یال x1 متصل است.

و . . .

tn یالی است که از یک طرف به یال x(n-1) و از طرفی دیگر به y متصل است.

اگرT= (t1,t2, . . . ,tn) نشان دهنده کوتاهترین مسیر بین x وy باشد آنگاه T در خود عضو تکراری ندارد، بنابراین اگر در T دو عضو یکسان ti و tj را داشته باشیم این مسیر کوتاهترین مسیر نیست.

گفتیم که اگر T مسیری بین x و y باشد آنگاه tn متصل به y است بنابراین اگر Y={y1,. . . ,yn} مجموعه ای شامل یالهای متصل به راس y باشد حتما tn Є Y است، از این موضوع برای خاتمه دادن به محاسابت استفاده می کنیم.

مجمو عه ای شامل یالهای متصل به راس x تشکیل داده و آنرا X می نامیم.

اولین عضو مجموعه X را انتخاب می کنیم، این کار به این معنی است که روی یالی که با اولین عضو مجموعه نشان داده می شود حرکت می کنیم.

t1 Є X

فرض کنیم که اولین عضو X (x1)یال ِ ai باشد با این فرض اولین عضو T برابر ai خواهد بود:

t1=ai

می توان از این یال به سمت یک یال که عضو Ai است حرکت کرد ، چون نباید از یک مسیر بیش از یکبار عبور کرد بنابراین یال دوم نمی تواند خود ai باشد:

t2 Є Ai-{t1}

فرض کنیم:

t2=aj

بنابراین:

t3 Є Aj-{t1,t2}

به همین ترتیب به محاسبات ادامه می دهیم تا به دو شرط برسیم:

1. Ti عضوی از Y باشد
2. Ax-{t1,t2, . . .} یک مجموعه تهی باشد.

در صورت تحقق شرط 1 یکی از مسیرهای ممکن بدست می آید.

در صورتیکه شرط 2 تحقق پیدا کند مسیر ما به y ختم نمی شود و جزو جوابهای ما نخواهد بود.

با این تفاسیر ما فقط مسیر T=(t1,t2, . . . ,tn) را بدست آوردیم، برای بدست آوردن تمام مسیرها باید در هر مرحله از یک حلقه استفاده کنیم که که تعداد تکرار این حلقه برابر تعداد اعضای مجموعه Ax-{t1,t2, . . .} در آن مرحله است.

پس از بدست آوردن تمام مسیرهای بدون تکرار بین x وy، کوتاه ترین مسیر که برابر با کوچکترین n تایی در بین T ها است را بدست می آوریم.

الگوریتم برنامه:

در ابتدا باید به پارامترهای تابع مقدار بدهید.

چون در ابتدا T عضوی در خود ندارد پس به T مقدار اولیه 0 یا همان تهی می دهیم.

F(x,y,T,A1,A2, . . . ,AN)

{

MAX <- NUMBER OF MEMBER OF X

COUNT=0

START:

COUNT= COUNT+1

TEMP <- X(COUNT)

add TEMP to end of T

If TEMP is member of Y then

print T

else

B= Ax that mapped to TEMP

B= B-T

If B=0

print T

Else

F(B,y,T,A1,A2, . . . ,AN)

If COUNT<>MAX then Goto START:

}

با گذاشتن چند شرط دیگر در داخل الگوریتم می توانید تعدادی دیگر از مسیرها را که جزء جواب نخواهند بود در همان مرحله محاسبات حذف کرد.

مسیرهایی که در این الگوریتم بدست می آیند ممکن است به y ختم نشوند و یا ممکن است که به y ختم شوند ولی کوتاهترین مسیر نباشند، کاری که شما باید انجام دهید این است که با گذاشتن شرط این مسیرها را حذف کنید، من برای اینکه الگوریتم پچیده نشود این کار را انجام ندادم.

لطفا نظرتون رو در مورد این الگوریتم به من بفرستید.

Yaser.hafizi@gmail.com