

Анализ на задача *police*

Тагове: *DFS*, артикулационни ребра (мостове), двоично повдигане, двоично търсене

Решение за 23 точки (30%)

Достатъчно е да симулираме блокирането на връх и ребро и да пуснем *DFS* от x , за да видим дали можем да достигнем y . Блокирането на връх може да стане просто с маркирането му като посетен, а блокирането на ребро като например допълнително маркираме двата върха и следим да не използваме ребро между два маркирани върха.

Сложност: $O(Q(N + M))$.

Пълно решение за 75 точки (100%)

Нека разгледаме по-простия случай - когато се блокира ребро. Първо трябва да проверим дали то въобще съществува. Това може да стане по много начини, но ще опишем този, който използва информация нужна и по-нататък. Ако въведем фиктивен корен, то можем да намерим *in* и *out* времето на всеки връх. Нека a е потенциалният баща, а b е потенциалният син (слагаме първия връх да е този с по-малкото *in* време). Остава да се уверим, че b е в поддървото на a и дълбочината му е с 1 по-голяма. Вече можем да считаме, че сме проверили за съществуването на реброто. Ако то не е артикулационно, премахването му няма да промени свързаността в графа, така че отговорът е да. В противен случай, за да остане свързаността на двата върха трябва те едновременно да са наследници на b или едновременно предшественици на a . Ако това не е така, то те са от двете страни на реброто и премахването му ще раздели двата върха в различни свързани компоненти. Повече за артикулационните върхове и ребра може да прочетете [тук](#).

За да се ориентираме в другия случай, трябва да видим къде се намират двата върха спрямо a . За тази цел ни трябва съседите на a , през които минава пътя до x и y в покриващото дърво на *DFS*-а. Ако се ползва един и същ съсед, то блокирайки връх a , няма да развалим свързаността между двата върха. Когато са различни съседите, то свързаността може да се развали единствено ако някое от двете ребра (между съседите и a) е артикулационно. Остана да изясним как намираме тези съседи на a . Нека фиксираме връх x . Ако той е предшественик на връх a , то търсеният съсед е просто бащата на a . По-интересен е случаят, когато x е наследник на a . Един начин е да правим двоично търсене по интервалите от *in* и *out* времена на децата на a , за да намерим в кой интервал попада *in* времето на връх x . Авторът използва друг начин - чрез двоично повдигане, изкачваме връх x на съответната дълбочина нагоре, така че да стане с дълбочина 1 повече от a .

Сложност: $O(N + M + Q \log_2 N)$.

Автор: Илиян Йорданов (заета)