Zadanie: SPO
Spotkanie towarzyskie

Warsztaty ILO 2017-2018, grupa olimpijska, dzień 2. Dostępna pamięć: 128 MB.

Rozważmy graf, w którym wierzchołki odpowiadają znajomym Przemka (analogicznie jak w drzewie na wejściu), natomiast dwóch znajomych jest połączonych krawędzią o wadze c wtedy i tylko wtedy, gdy istnieje linia telefoniczna o koszcie c, do której należą obaj znajomi. Innymi słowy, mogą się oni tym kosztem skomunikować.

Wówczas okazuje się, że w naszym zadaniu szukamy minimalnego drzewa rozpinającego tego grafu, a dokładniej spójnej składowej (jeśli graf nie jest spójny), w której znajduje się Przemek. Jest tak dlatego, że taka spójna wyznacza nam wszystkich znajomych, których Przemek jest w stanie zaprosić (niezależnie od kosztu), natomiast suma wag krawędzi w tym drzewie rozpinającym jest sumarycznym kosztem zaproszenia tych znajomych (wybrane krawędzie wyznaczają nam, kto kogo zaprasza). W związku z tym, minimalizacja tej sumy wag jest rozwiązaniem naszego zadania, a to właśnie z definicji jest minimalne drzewo rozpinające.

Obserwacja.1. *Minimalne drzewo rozpinające możemy znaleźć algorytmem, który zachłannie wybiera krawędzie najtańsze.*

Dowód tej obserwacji nie będzie tu przedstawiony, ponieważ jest to bardzo standardowy algorytm (algorytm Kruskala).

Niestety, wykorzystanie algorytmu Kruskala, pomimo jego szybkości, jest w tym przypadku nieefektywne, ponieważ sam graf może mieć kwadratowy rozmiar względem rozmiaru wejścia. Powyższe rozumowanie miało jedynie wskazać, że najlepiej jest wybierać zachłannie najtańsze linie telefoniczne.

Rozwiązanie wzorcowe $O(m\ log\ m + n\ log^*\ n)$

Rozwiązanie będzie wyglądało następująco. Ukorzeniamy sobie drzewo w dowolnym wierzchołku oraz sortujemy wszystkie linie telefoniczne względem ich wag niemalejąco. Dla każdego wierzchołka w drzewie będziemy utrzymywać strukturę pamiętającą najniższego jego przodka (tj. o najmniejszej głębokości), z którym dany wierzchołek jest już aktualnie połączony pewnymi liniami telefonicznymi. Na początku wszystkie wierzchołki wskazują na samego siebie.

Następnie przetwarzamy w ustalonej kolejności linie telefoniczne i znajdujemy ich końce (dwa wierzchołki będące końcami pewnej ścieżki). Następnie dla każdego z nich, skaczemy w górę drzewa korzystając z naszej struktury i scalamy kolejne wierzchołki z ich ojcami naszą nową linią telefoniczną tak długo, aż wskaźniki z obu wierzchołków spotkają się w jednym punkcie.

Oba wskaźniki powinny się spotkać w najniższym wspólnym przodku wierzchołków będących końcami ścieżki lub w wierzchołku odrobinę wyżej, który jest z tym przodkiem już połączony.

Strukturę, która umożliwia nam efektywne skakanie w górę (chcemy omijać krawędzie już do tej pory złączone pewną linią), możemy zaimplementować używając Find&Union. Reprezentant każdego zbioru to wierzchołek, który jest najniżej, z którym wszyscy są połączeni. Jeśli scalamy pewien wierzchołek z jego ojcem w drzewie, to reprezentantem syna staje się reprezentant ojca. Dodatkowo, musimy utrzymywać koszt uzyskania danego zbioru. Zatem gdy łączymy wierzchołek z jego ojcem krawędzią, to dodatkowo dodajemy do kosztu tego zbioru wartość aktualnie przetwarzanej linii telefonicznej.

Łatwo zauważyć, że aby zwrócić wynik, obliczamy ile wierzchołków ma tego samego reprezentanta, co wierzchołek, którym jest Przemek oraz sprawdzamy, jaki był koszt uzyskania zbioru, w którym on jest.

Złożoność składa się z dwóch członów. $m\ log\ m$ oznacza posortowanie linii telefonicznych, a $n\ log^*\ n$ to złożoność Find&Union.

Fundusze Europejskie Program Regionalny
Podlaskie
Unia Europejska Europejski Fundusz Społeczny