8 stycznia 2020

Zadania z matematyki dyskretnej, lista nr 12

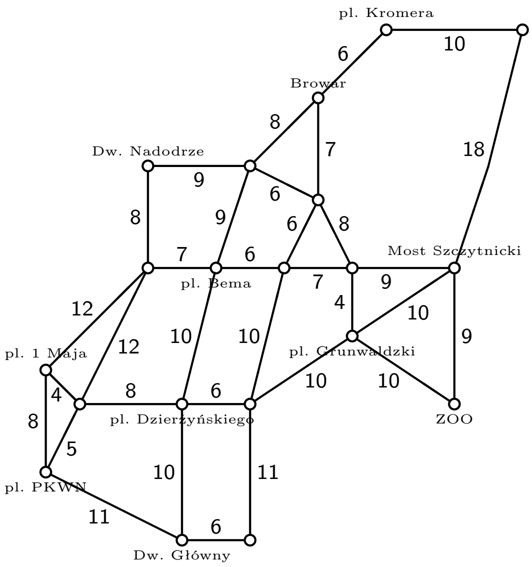
1. Zastosuj przeszukiwanie grafu spójnego w głąb do znajdowania wierzchołka *rozcinającego*, tzn. takiego którego usunięcie rozspaja graf (jeśli taki wierzchołek istnieje). Twoja procedura powinna mieć złożoność $O(m+n)$.

2. Zastosuj przeszukiwanie grafu w głąb do sprawdzania, czy graf jest dwudzielny. Twoja procedura powinna mieć złożoność $O(m+n)$.

3. *Topologiczne porządkowanie wierzchołków*. Niech G będzie digrafem acyklicznym (bez skierowanych cykli). Napisz procedurę, która w czasie $O(n+m)$ przyporządkowuje numery wierzchołkom w taki sposób, że gdy (i,j) jest łukiem w G, to $i<j$.

4. Używając algorytmów Prima–Dijkstry i Kruskala, wyznacz najkrótszy podzbiór dróg z Rysunku poniżej taki, że po tym podzbiorze dróg można dojechać z każdego węzła tej sieci do każdego innego. Ponumeruj krawędzie tego podzbioru w kolejności, w jakiej dołączają je do niego oba te algorytmy.

5. Używając algorytmu Dijkstry wyznacz drzewa najkrótszych dróg z Dw. Głównego, pl. Grunwaldzkiego i Browaru w sieci z Rys. 1. Ponumeruj krawędzie drzew w kolejności dołączania przez algorytm.



6. (Dualny algorytm chciwości) Udowodnij, że po wykonaniu kroków 1 i 2, T jest najkrótszym drzewem rozpinającym grafu G.

   Krok 1: Niech $c(e_1) > c(e_2) > ... > c(e_m)$;
   Krok 2: $T := E(G)$;
   `for` $e_1, e_2, ..., e_m$ `do`
   `if` $T \setminus e_i$ jest grafem spójnym
   `then` $T := T \setminus e_i$.

7. Pokaż, w jaki sposób można znaleźć najdłuższe drzewo rozpinające grafu z wagami.

8. Podaj szybką implementację algorytmu Warshalla dla obliczania przechodniego domknięcia grafu o 32 wierzchołkach. Macierz sąsiedztwa zawarta jest w postaci tablicy 32 słów 32–bitowych reprezentujących kolejne wiersze. Możesz wykonywać operacje logiczne na słowach 32 bitowych.

9. Zmodyfikuj tak algorytm Warshalla-Floyda, aby znajdował nie tylko długości najkrótszych drogi między parami wierzchołków, ale również te drogi.

10. W digrafie może istnieć wiele minimalnych dróg z wierzchołka v do u. Zmodyfikuj tak algorytm Dijkstry, żeby wyznaczał nie drzewo najkrótszych dróg, a acykliczny digraf będący sumą najkrótszych dróg z wierzchołka v do wszystkich innych.

11. Drzewo z dodaną jedną krawędzią nazywamy 1-drzewem. Zaproponuj (działający w rozsądnym czasie) algorytm znajdujący najkrótsze 1-drzewo spinające grafu z wagami i uzasadnij jego poprawność.

12. Niech G będzie grafem z nieujemnymi wagami na krawędziach. Niech $\mathrm{MST}(G)$ oznacza długość najlżejszego drzewa spinającego w G, a $\mathrm{TSP}(G)$ oznacza długość najkrótszej drogi komiwojażera w G (komiwojażer może odwiedzać wierzchołki wielokrotnie). Wykaż, że
$$\mathrm{MST}(G) \leq \mathrm{TSP}(G) \leq 2\cdot \mathrm{MST}(G).$$

13. Dana jest sieć, w której każda krawędź poza pojemnością ma określony minimalny przepływ dla każdego łuku. Dany mamy pewien dopuszczalny przepływ f w tej sieci. Pokaż jak wyznaczyć maksymalny i minimalny przepływ w tej sieci.

14. (Twierdzenie Mengera) Graf jest krawędziowo k-spójny gdy jest spójny i usunięcie z niego co najwyżej $k-1$ krawędzi nie rozspójnia go. Używając przepływów w sieciach pokaż, że G jest krawędziowo k spójny wtedy i tylko wtedy gdy między każdymi dwoma wierzchołkami istnieje k krawędziowo rozłącznych dróg.

15. (Wersja wierzchołkowa Twierdzenia Mengera) Graf jest k-spójny gdy jest spójny i usunięcie z niego co najwyżej $k-1$ wierzchołków nie rozspójnia go. Pokaż, że G jest k-spójny wtedy i tylko wtedy gdy między każdymi dwoma niesąsiednimi wierzchołkami istnieje k wierzchołkowo rozłącznych dróg.