Algorytmy i Struktury Danych

7 maja 2021

Ćwiczenia 8: Grafy, BFS i DFS

Zadania obowiązkowe

Zadanie 1. (Pause) Znany operator telefonii komórkowej Pause postanowił zakonczyc działalnosc w Polsce. Jednym z głównych elementów całej procedury jest wyłaczenie wszystkich stacji nadawczych (które tworza spójny graf połaczen). Ze wzgledów technologicznych urzadzenia nalezy wyłaczac pojedynczo a operatorowi dodatkowo zalezy na tym, by podczas całego procesu wszyscy abonenci znajdujacy sie w zasiegu działajacych stacji mogli sie ze soba łaczyc (czyli by graf pozostał spójny). Proszę zaproponować algorytm podający kolejność wyłączania stacji.

Zadanie 2. (cykl na cztery) Dany jest graf nieskierowany G zawierający n wierzchołków. Zaproponuj algorytm, który stwierdza czy w G istnieje cykl składający się z dokładnie 4 wierzchołków. Zakładamy, że graf reprezentowany jest przez macierz sasiedztwa A.

Zadania standardowe

Zadanie 1. (DFS/BFS) Proszę zaimplementować następujące algorytmy:

1. Sprawdzanie czy graf jest dwudzielny (czyli zauważyć, że to 2-kolorowanie i użyć DFS lub BFS).
2. Policzyć liczbę spójnych składowych w grafie (implementacja przeciwna do tej z poprzedniego zadania)

Zadanie 2. (uniwersalne ujście) Mówimy, że wierzchołek t w grafie skierowanym jest uniwersalnym ujściem, jeśli (a) z każdego innego wierzchołka v istnieje krawędź z v do t, oraz (b) nie istnieje żadna krawędź wychodząca z t.

1. Podać algorytm znajdujący uniwersalne ujście (jeśli istnieje) przy reprezentacji macierzowej ($O(n^2)$).
2. Pokazać, że ten problem można rozwiazac w czasie $O(n)$ w reprezentacji macierzowej.

Zadanie 3. (BFS i najkrótsze ścieżki) Proszę zaimplementować algorytm BFS tak, żeby znajdował najkrótsze ścieżki w grafie i następnie, żeby dało się wypisać najkrotszą ścieżkę z zadanego punktu startowego do wskazanego wierzchołka.

Zadanie 4. (malejące krawędzie) Dany jest graf $G = (V, E)$, gdzie każda krawędź ma wagę ze zbioru $\{1, \dots, |E|\}$ (wagi krawędzi są parami różne). Proszę zaproponować algorytm, który dla danych wierzchołków x i y sprawdza, czy istnieje ścieżka z x do y, w której przechodzimy po krawędziach o coraz mniejszych wagach.

Zadanie 5. (krawędzie 0/1) Dana jest mapa kraju w postaci grafu $G = (V, E)$. Kierowca chce przejechać z miasta (wierzchołka) s to miasta t. Niestety niektóre drogi (krawędzie) są płatne. Każda droga ma taką samą jednostkową opłatę. Proszę podać algorytm, który znajduje trasę wymagającą jak najmniejszej liczby opłat. W ogólności graf G jest skierowany, ale można najpierw wskazać algorytm dla grafu nieskierowanego.

Zadanie 6. (bezpieczny przelot) Dany jest graf $G = (V, E)$, którego wierzchołki reprezentują punkty nawigacyjne nad Bajtocją, a krawędzie reprezentują korytarze powietrzne między tymi punktami. Każdy korytarz powietrzny $e_i \in E$ powiązany jest z optymalnym pułapem przelotu $p_i \in \mathbb{N}$ (wyrażonym w metrach). Przepisy dopuszczają przelot danym korytarzem jeśli pułap samolotu różni się od optymalnego najwyżej o t metrów. Proszę zaproponować algorytm (bez implementacji), który sprawdza czy istnieje możliwość przelotu z zadanego punktu $x \in V$ do zadanego punktu $y \in V$ w taki sposób, żeby samolot nigdy nie zmieniał pułapu. Algorytm powinien być poprawny i możliwie jak najszybszy. Proszę oszacować jego złożoność czasową.

Zadanie 7. (kosztowna szachownica) Dana jest szachownica o wymiarach $n \times n$. Każde pole (i, j) ma koszt (liczbę ze zbioru $\{1, \ldots, 5\}$) umieszczony w tablicy A (na polu $A[j][i]$). W lewym górnym rogu szachownicy stoi król, którego zadaniem jest przejsc do prawego dolnego rogu, przechodzac po polach o minmalnym sumarycznym koszcie. Prosze zaimplementowac funkcje `kings_path(A)`, która oblicza koszt sciezki króla. Funkcja powinna byc mozliwie jak najszybsza.

Jak zostanie czas i trzeba czymś wypełnić

Zadanie 1. (kapitan statku, zadanie z kolokwium w 2012/13) Kapitan pewnego statku zastanawia się, czy może wpłynąć do portu mimo, że nastąpił odpływ. Do dyspozycji ma mapę zatoki w postaci tablicy M, gdzie $M[y][x]$ to głebokość zatoki na pozycji (x, y). Jeśli jest ona większa niż pewna wartość `int T` to statek może się tam znaleźć. Początkowo statek jest na pozycji $(0, 0)$ a port znajduje się na pozycji $(n-1, m-1)$. Z danej pozycji statek może przepłynąć bezpośrednio jedynie na pozycję bezpośrednio obok (to znaczy, na pozycję, której dokładnie jedna ze współrzędnych różni się o jeden). Proszę napisać funkcję rozwiązującą problem kapitana.

Zadanie 2. (czy nieskierowany?) Proszę podać algorytm, który mając na wejściu graf G reprezentowany przez listy sąsiedztwa sprawdza, czy jest nieskierowany (czyli czy dla każdej krawędzie $u \to v$ istnieje także krawędź przeciwna).