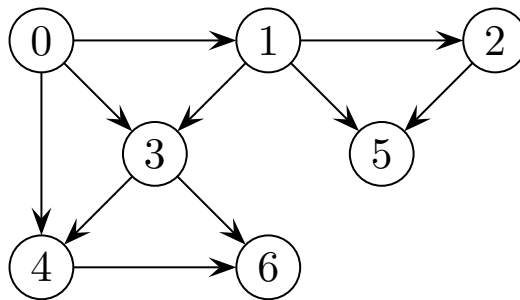


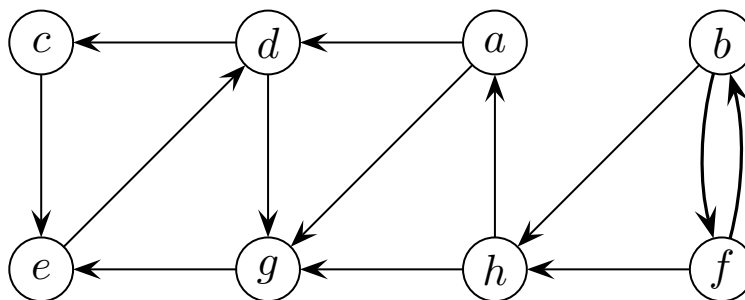
Przeszukiwanie grafów

Zadanie AiSD.L.Graph.01



- Podaj macierz sąsiedztwa A i listę sąsiedztw L dla podanego wyżej grafu skierowanego.
- Wypisz, w jakiej kolejności będą odwiedzone wierzchołki w czasie przeszukiwania tego grafu w głąb z wierzchołkiem startowym 0. Z jakich krawędzi zbudowane jest wynikowe drzewo DFS?
- Wypisz, w jakiej kolejności będą odwiedzone wierzchołki w czasie przeszukiwania tego grafu wszerz z wierzchołkiem startowym 0. Z jakich krawędzi zbudowane jest wynikowe drzewo BFS?

Zadanie AiSD.L.Graph.02



Zakładając, że wierzchołki są przeglądane w porządku alfabetycznym, a każda lista sąsiedztwa jest uporządkowana alfabetycznie, zilustruj działanie algorytmu przeszukiwania w głąb dla grafu z powyższego rysunku zaczynając od wierzchołka o etykiecie (a) a oraz (b) d . Podaj dla każdego wierzchołka czas jego odwiedzenia i przetworzenia, a także sklasyfikuj każdą krawędź.

Zadanie AiSD.L.Graph.03 (5 pkt.)

Zaimplementuj omawiane na wykładzie algorytmy przeszukiwania grafów.

- DFS, czyli przeszukiwanie w głąb (2 pkt.);
- BFS, czyli przeszukiwanie wszerz (2 pkt.);
- Dodatkowy 1 punkt, jeżeli po wczytaniu z pliku graf przechowywany jest w postaci listy sąsiedztw.

Specyfikacja wejścia/wyjścia.

Wejście. Skierowany graf $G = (V, E)$ w pliku w postaci macierzy sąsiedztwa $A = [x_{ij}]_{N \times N}$. Pierwszy wiersz pliku zawiera liczb N wierzchołków grafu. Każdy kolejny $(i + 1)$ -szy wiersz w pliku zawiera ciąg N liczb $x_{i1} x_{i2} \cdots x_{iN}$ tworzących i -ty wiersz macierzy A .

Wyjście. Wierzchołki uporządkowane według pierwszego odwiedzenia przez dany algorytm przeszukiwania grafu; dodatkowo krawędzie otrzymanego lasu spinającego.

Uwaga. Przy demonstrowaniu programu należy pokazać rysunki testowanych grafów (przynajmniej trzy, np. na kartce) i wykazać się rozumieniem zaimplementowanych algorytmów.