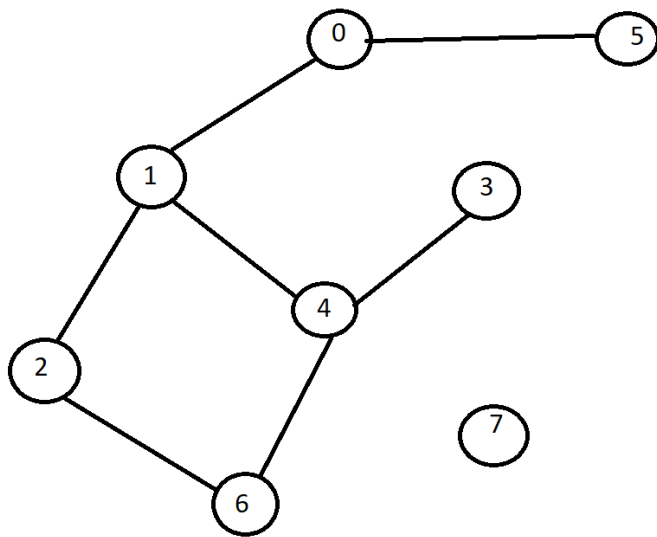


Przeszukiwanie w głąb (DFS)



- 1) Rozpoczynamy od zadanego wierzchołka (dla ustalenia uwagi weźmy wierzchołek nr 1, jest to tzw. wierzchołek startowy). Będzie to wierzchołek „aktualny”. Oznaczamy go jako odwiedzony. Dodajemy do listy wyników.
- 2) Patrzymy na jego sąsiadów, wybieramy tego, który jest jeszcze nieodwiedzony i ma najniższy indeks. Przechodzimy do wybranego wierzchołka (będzie to nowy wierzchołek aktualny, oznaczamy go jako odwiedzony i wpisujemy na listę wyników).
- 3) Ponawiamy krok 2 dla nowego aktualnego wierzchołka.
- 4) Jeśli aktualny wierzchołek nie ma nieodwiedzonych sąsiadów to ponawiamy krok 2 dla jego rodzica.
- 5) Gdy wrócimy do wierzchołka startowego i nie ma on nieodwiedzonych sąsiadów to algorytm się kończy.

Przykładowo gdy zaczynamy od 1 to wynik będzie taki:

1 0 5 2 6 4 3

Dlaczego? W pierwszym kroku (aktualnym wierzchołkiem była 1) było dwóch nieodwiedzonych sąsiadów : 2 oraz 0, 0 jest mniejsze więc było nowym aktualnym (dodajemy do wyników). Z niego przeszło się do 5 (znów dodajemy). Piątka nie miała żadnych nieodwiedzonych sąsiadów więc trzeba było wrócić do 0, ono też już nie miało nieodwiedzonych sąsiadów więc wracamy do 1. Dla 1 nieodwiedzonym sąsiadem była 2 (ją dodajemy do listy wyników). Z 2 idziemy do 6 potem do 4 i do 3. Później są tylko powroty, finalnie do 1, która już nie ma nieodwiedzonych sąsiadów.

Wskazówki:

Implementacja np. przez rekurencyjne wywoływanie funkcji.

