# Sprawozdanie nr. 3

# Grafy

Autorzy:

Mateusz Babiaczyk, 127325

Rafał Budka, 127269

1. Złożoności reprezentacji grafu:
2. Tabela krawędzi:

Jedna z gorszych reprezentacji grafu. Jest to tabela o wielkości równej ilości łuków w grafie. Dla każdego łuku mamy podane wierzchołki z nim incydentne.

Jej złożoność pamięciowa to O(m), gdzie m jest liczbą łuków w grafie i a więc przy grafie pełnym może to być nawet (n\*(n-1))/2 (n – liczba wierzchołków w grafie).

Złożoność znalezienia jednej krawędzi wynosi również O(m) gdyż musimy przeszukać całą tablice krawędzi by znaleźć dany wierzchołek i odczytać jego następnika, tak samo jest w przypadku wyszukiwania wszystkich możliwych następników.

1. Macierz sąsiedztwa:

Macierz jest przedstawiona w postaci tablicy dwuwymiarowej o wielkości n x n (n – ilość wierzchołków w grafie). Tablica jest wypełniona zerami i jedynkami, gdzie odpowiednio, zero oznacza, że dwa wierzchołki nie mają połączenia, a jedynka, że istnieje takie połączenie.

Złożoność pamięciowa tej reprezentacji wynosi O(n2), z uwagi iż mamy tablice dwuwymiarową o wielkości n x n.

Złożoność wyszukania następnika jak i wyszukania wszystkich następników wynosi O(n), gdyż przy poszukiwaniu jednego następnika w najgorszym przypadku musimy przejść przez wszystkie możliwe wierzchołki w poszukiwaniu „jedynki”, a przy wyszukiwaniu wszystkich możliwych następników również musimy sprawdzić wszystkie wierzchołki.

1. Lista sąsiedztwa następników:

Jedna z lepszych form przedstawienia grafu. Jest to lista o długości n (n – ilość wierzchołków), z podlistami o długości ilości następników danego wierzchołka czyli może to być liczba od 0 do n.

Złożoność pamięciowa tej listy wynosi O(n+m), (n – ilość wierzchołków, m – ilość krawędzi), gdyż mamy długość n wierzchołków i m krawędzi zapisanych jako następnicy.

Złożoność wyszukiwania następnika wynosi O(1), czyli jest to złożoność stała. Jest tak z uwagi na to że po wejściu do odpowiedniego wierzchołka jej pierwsza pozycja na liście to następnik, ewentualnie lista będzie pusta i nie będzie taki wierzchołek posiadał następników.

Pesymistyczna złożoność wyszukiwania wszystkich następników wynosi O(n), gdyż musimy przejrzeć całą listę danego wierzchołka. Jednak średnio złożoność ta wynosi O(m/n) gdyż tyle średnio następników znajduje się w danym wierzchołku.

1. Macierz grafu:

Tworzy się ją na podstawie listy następników, listy poprzedników i listy braku incydencji. Jest to tablica dwuwymiarowa n x n z dodatkową tabelą o wymiarach n x 3. Mniejsza tabela wypełniona jest po kolei idąc: pierwszym następnikiem, pierwszym poprzednikiem, pierwszym wierzchołkiem z listy braku incydencji. Jeżeli nie ma żadnego wierzchołka w którejś z list wpisujemy tam zero. W większej tabeli wpisujemy posiłkując się listami wierzchołki i kończąc uzupełnianie danego rodzaju poprzez wpisanie takiego samego numeru jak aktualnie wskazany wierzchołek. Poprzedniki zapisujemy pomnożone przez (-1) a braku incydencji poprzez dodanie n (n – ilość wierzchołów). Następniki zawierają więc liczby od 1 do n, poprzedniki od -n do -1, a brak incydencji od n+1 do n+n.

Złożoność pamięciowa wynosi O(n\*(n+3)) gdyż mamy tabelę o wielkości n x (n+3).

Złożoność wyszukiwania następnika wynosi O(1), gdyż po wejściu w dany wierzchołek wiemy od którego zacząć dzięki dodatkowej tabeli i nie musimy przeszukiwać wszystkich pól w wierszu tabeli.

Złożoność wyszukiwania wszystkich następników wynosi O(m/n) gdyż przeszukujemy tylko następniki które wskazują nam kolejne.

1. Złożoność obliczeniowa algorytmów sortowania topologicznego:

Sortować można tylko grafy skierowane, acykliczne.

1. Sortowanie topologiczne poprzez usuwanie wierzchołków o stopniu wchodzącym 0:

Sortowanie to działa poprzez wyszukiwanie wierzchołków niezależnych czyli o stopniu wejściowym równym zero wraz z wychodzącymi z nich krawędziami. Przy rodzaju reprezentacji wszystko zależy tylko od rodzaju zapisu tych krawędzi. Im mniej przeszukiwania tym szybciej działa algorytm.

Jak widać powyżej najlepszy czas posiada lista następników oraz macierz grafu gdyż algorytm sprawdza wszystkich następników, a te dwie reprezentacje mają najmniejszą złożoność tego wyszukiwania równą średnio O(m/n). Więc złożoność obliczeniowa algorytmu wynosi średnio O(m), gdyż zajmuje nam to n \* (m/n) = m.

Gorszy czas posiada już macierz sąsiedztwa z uwagi na to że wyszukiwanie wszystkich następników zajmuje mu O(n), przez co za każdym razem musi przejrzeć większą ilość pól niż jego poprzednicy. Więc złożoność obliczeniowa algorytmu korzystając z tej reprezentacji wynosi O(n2). (Gdyż n\*n).

Najgorzej wypada Tabela krawędzi której czas wyszukiwania wszystkich następników wynosi O(m). A więc złożoność obliczeniowa algorytmu dla tabeli krawędzi wynosi O(n\*m), co jest naprawdę słabym czasem co widać na 2 wykresie.

1. Sortowanie topologiczne poprzez wykorzystanie algorytmu DFS:

Sortowanie topologiczne polega na stworzeniu listy wierzchołków, w której każdy wierzchołek posiadający następniki znalazł się przed nimi. Przejście grafu metodą DFS (w głąb – deep first search) to drugi sposób toposort’a na grafach. Algorytm umieszcza wierzchołki na stosie po rekurencyjnym umieszczeniu tam wszystkich jego następników. Jeżeli podczas przechodzenia grafu metodą DFS skończą się możliwości dotarcia do nieodwiedzonych wierzchołków algorytm przerywa się i wznawia od losowo wybranego nieodwiedzonego wierzchołka.

Na załączonych wykresach widać, że czas sortowania grafu dla macierzy grafu porównywalna z listą następników. Wynika to z faktu złożoności czasowej wyszukiwania wszystkich następników w macierzy grafu, która wynosi średnio O(m/n), czyli dla wszystkich n wierzchołków wynosi O(m). Z kolei złożoność obliczeniowa wyszukiwania wszystkich następników wierzchołka w liście sąsiedztwa wynosi w pesymistycznym przypadku O(n) jeśli wszystkie wierzchołki oprócz badanego są jego następnikami. Średnia złożoność wynosi również O(m/n) co po pomnożeniu razy ilość wierzchołków do przetworzenia daje nam złożoności od O(m) do O(n2). Dlatego macierz grafu jest bardziej optymalna, ponieważ zachowuje się podobnie dla wszystkich przypadków (nie ma przypadków pesymistycznych i optymistycznych).

Słabiej wypadła macierz sąsiedztwa, ponieważ wyszukiwanie wszystkich następników dla jednego wierzchołka wynosi O(n) co dla wszystkich wierzchołków daje nam czas O(n2 ).

Najsłabiej, tak samo jak w BFSie wypadła tabela krawędzi. Wynika to ze złożoności czasowej na znalezienie wszystkich następników wynoszącej O(m\*n) co dla wszystkich n wierzchołków daje nam O(n2 \* m).