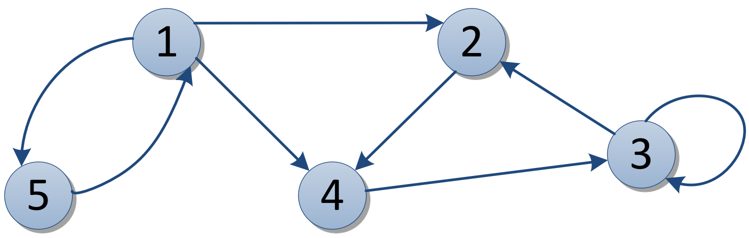
STRUKTURA DANYCH - GRAF

1. Grafy – wprowadzenie.

Graf $G = (V, E)$ jest strukturą składającą się ze zbioru wierzchołków (węzłów) i zbioru krawędzi łączących wierzchołki ze sobą. **Zbiór wierzchołków** będziemy oznaczać przez V (od ang. *vertex*), a **zbiór krawędzi** przez E (ang. *edge*).

Najbardziej intuicyjną reprezentacją grafu jest reprezentacja graficzna. Węzły reprezentowane są najczęściej jako koła, natomiast krawędzie jako linie (w grafach nieskierowanych) lub strzałki (w grafach skierowanych) łączące ze sobą węzły. Poniższy rysunek przedstawia **graf skierowany** składający się z 5 wierzchołków i 8 krawędzi:



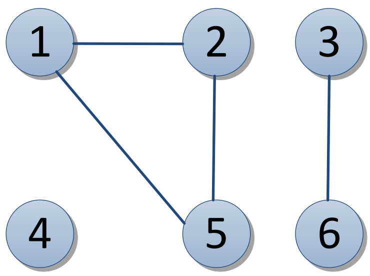
Rys. 1

Ten graf jest wyznaczony przez zbiór wierzchołków V = {1, 2, 3, 4, 5} oraz następujący zbiór krawędzi: E = { (1, 5), (5, 1), (1, 4), (1, 2), (2, 4), (4, 3), (3, 2), (3, 3) }. Zwróćmy uwagę, że krawędź jest wyznaczona przez parę wierzchołków, a zbiór krawędzi to zbiór par wierzchołków. Elementy pary są uporządkowane.

W grafie skierowanym możliwe jest istnienie **pętli** od danego wierzchołka do niego samego.

Jeśli (u, v) jest krawędzią grafu skierowanego, to mówimy, że krawędź (u, v) jest **wychodząca** z wierzchołka u i jest **wchodząca** do wierzchołka v.

W **grafie nieskierowanym** zbiór krawędzi E to zbiór *nieuporządkowanych* par wierzchołków. Krawędzie (u, v) i (v, u) oznaczają tę samą krawędź. W grafie nieskierowanym nie mogą występować pętle.



Rys. 2

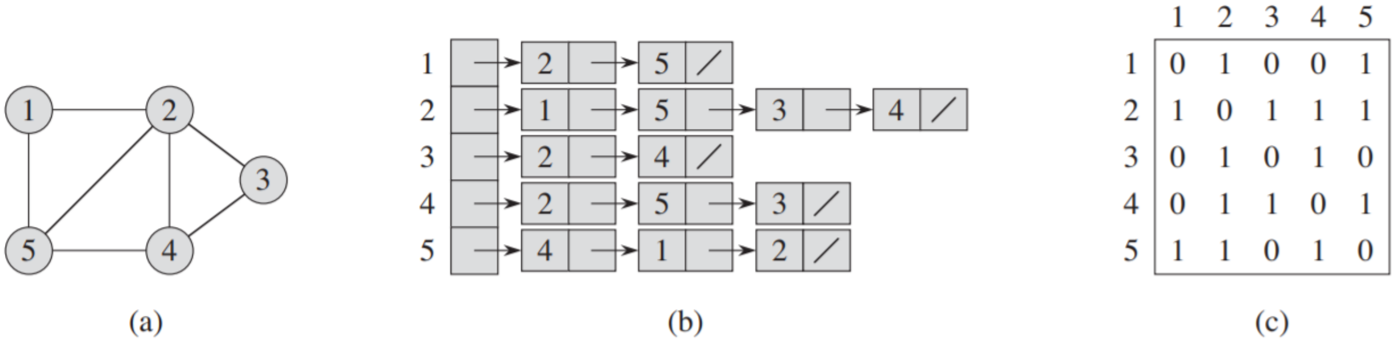
Stopniem wierzchołka w grafie *nieskierowanym* jest liczba krawędzi łączących go z innymi węzłami. Np. wierzchołek 2 na Rys. 2 ma stopień 2. Wierzchołek, którego stopień wynosi 0, taki jak wierzchołek 4 na Rys. 2, jest nazywany **izolowanym**.

W grafie skierowanym **stopień wyjściowy** wierzchołka jest liczbą krawędzi wychodzących z niego, a **stopień wejściowy** jest liczbą krawędzi do niego wchodzących. **Stopniem wierzchołka** w grafie *skierowanym* jest liczba będąca sumą jego stopni: wejściowego i wyjściowego. Wierzchołek 3 na Rys. 1 ma stopień wejściowy 2, stopień wyjściowy 2 i stopień 4.

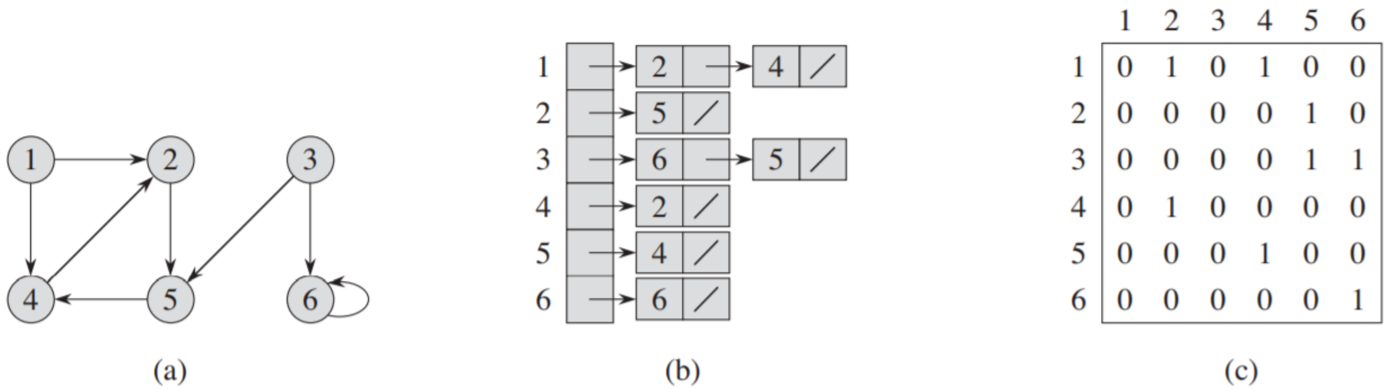
Ścieżka (droga) **długości** k z wierzchołka u do wierzchołka u' jest ciągiem wierzchołków $\langle v_0, v_1, \dots, v_k \rangle$ takich, że $u = v_0, u' = v_k$ i $(v_{i-1}, v_i) \in E$ dla $i = 1,2, \dots, k$. **Długość ścieżki** jest liczbą krawędzi. Jeśli istnieje ścieżka z u do u', to mówimy, że u' jest **osiągalny** z u jeśli G jest grafem skierowanym. Ścieżkę nazywamy **prostą**, jeśli wszystkie jej wierzchołki są różne. Na Rys. 1 ścieżka $\langle 5,1,2,4,3 \rangle$ jest ścieżką prostą o długości 4. Ścieżka $\langle 1,5,1,4,3 \rangle$ nie jest ścieżką prostą.

2. Reprezentacja grafów.

Istnieją dwa standardowe sposoby reprezentacji grafu $G = (V, E)$: za pomocą list sąsiedztwa lub macierzy sąsiedztwa. Reprezentacja listowa przedstawia w zwarty sposób **grafy rzadkie** - dla których liczba krawędzi $|E|$ jest dużo mniejsza od $|V|^2$.



Dwie reprezentacji grafu nieskierowanego. (**a**) Graf nieskierowany G z pięcioma wierzchołkami i siedmioma krawędziami. (**b**) Listy sąsiedztwa grafu G. (**c**) Macierz sąsiedztwa grafu G.



Dwie reprezentacji grafu skierowanego. (**a**) Graf skierowany G z sześcioma wierzchołkami i ośmioma krawędziami. (**b**) Listy sąsiedztwa grafu G. (**c**) Macierz sąsiedztwa grafu G.

Wyszukiwanie najkrótszej ścieżki

Algorytm Dijkstry

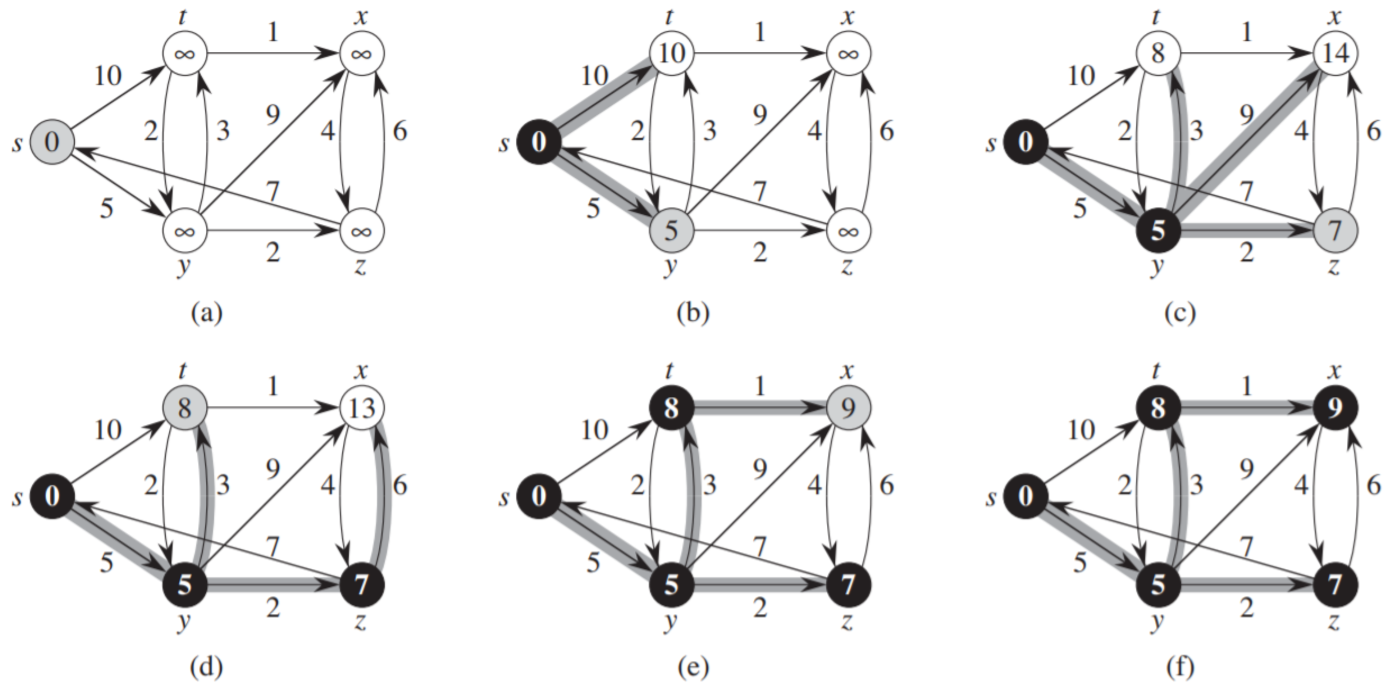
Problem: znalezienie najkrótszej ścieżki z danego wierzchołka do wszystkich pozostałych wierzchołów w grafie spójnym z wagami.

Reguła zachłanna: Spośród wszystkich wierzchołków, które mogą przedłużyć najkrótszą ścieżkę do tej pory znalezioną, wybierz ten, którego dodanie prowadzi dalej do najkrótszej ścieżki.

Wejście: $G = (V, E, w)$ – graf spójny z wagami, s - wierzchołek startowy (źródło).
Wyjście: dla każdego wierzchołka u osiągalnego z s obliczona odległość z s do u.

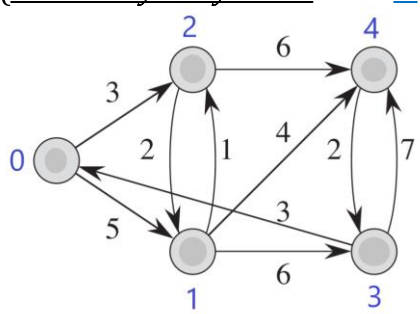
Algorytm:

for(każdy wierzchołek $u \in V$)
    ustaw odległość $d[u] = \infty$
    $p[u] = \text{NIL}$
$d[s] = 0$
utwórz kolejkę priorytetową Q (typu min) ze wszystkich wierzchołków $\in V$, w której wierzchołki są zorganizowane według wartości d
while(kolejka Q nie jest pusta)
    zwróć element u kolejki o najmniejszej wartości d oraz usuń go z kolejki
    for(każda krawędź $(u, v) \in E$)
        if($d[v] > d[u] + w(u, v)$)
            $d[v] = d[u] + w(u, v)$
            $p[v] = u$



Zadanie 1. Wykonaj analizę działania algorytmu Dijkstry dla powyższego grafu i wierzchołka startowego *s*.
(<u>*Pliki do wykorzystania*</u>: *Lab8_zadania.xlsx*, arkusz *zadanie_1*).

Zadanie 2. Wykonaj analizę działania algorytmu Dijkstry dla poniższej przedstawionego grafu. Najpierw dla wierzchołka startowego 0, następnie dla wierzchołka 3.
(<u>*Pliki do wykorzystania*</u>: *Lab8_zadania.xlsx*, arkusz *zadanie_2*).



Bibliografia

T. H. Cormen, Ch. E. Leiserson, R. L. Rivest: *Wprowadzenie do algorytmów*. WNT Warszawa 2012.