Digraf

13 maja 2017

Graf skierowany, digraf, digraf prosty

Definicja 1

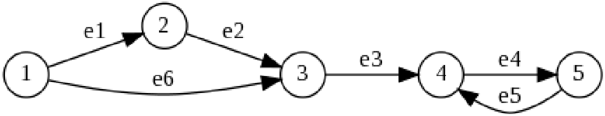
Digraf prosty G to (V, E), gdzie

V jest zbiorem *wierzchołków*,

E jest rodziną *zorientowanych krawędzi*, między różnymi wierzchołkami,
które nie mogą być wielokrotne.

Dokładniej E jest rodziną dwuelementowych **par** (v_1, v_2) różnych v_1, v_2 elementów V.

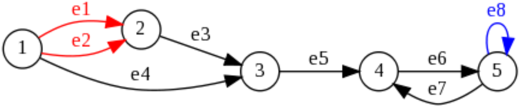
Zatem nie są dopuszczane skierowane pętle ani krawędzie wielokrotne jednakowo skierowane.


1
2
3
4
5
e1
e2
e3
e4
e5
e6

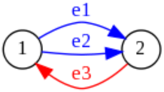
Multigraf skierowany

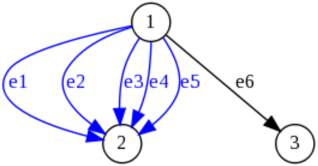
Multigraf skierowany M to $(V(G), E(G), \gamma : V(G) \times V(G) \to E(G))$.

Jeżeli e jest krawędzią multigrafu M, gdzie $\gamma(e) = (v_1, v_2)$,
to powiemy, że krawędź e wychodzi z wierzchołka v_1 do wierzchołka v_2
oraz, że e jest zorientowana z wierzchołka v_1 w kierunku wierzchołka v_2.


e1
e2
e3
e4
e5
e6
e7
e8
1
2
3
4
5


e3
e2
e1
1


e1
e2
e3
1
2


e1
e2
e3
e4
e5
e6
1
2
3

Multigraf skierowany

Uwaga 2

Digraf $(V(G), E(G), \gamma : V(G) \times V(G) \to E(G))$
to multigraf, w którym $\gamma : V(G) \times V(G) \to E(G))$ jest różnowartościowa.

Piszemy wtedy $e = (v, w)$ zamiast $\gamma(e) = (v, w)$.
Używa się też zapisu $e = \{v, w\}$, przy czym w tej notacji zbiór jest uporządkowany.

Multigraf skierowany

Jeśli $\gamma(e_1) = (v_1, v_2), \gamma(e_2) = (v_2, v_3)$, to powiemy, że krawędzie są zgodnie zorientowane w multigrafie.

W szczególności:

$\gamma(e_1) = (v_1, v_2), \gamma(e_2) = (v_2, v_1)$, są **zgodnie** zorientowane, ale nie tak **samo** zorientowane.

$\gamma(e_1) = (v_1, v_2), \gamma(e_3) = (v_1, v_2)$, są **nie zgodnie** zorientowane, ale są tak **samo** zorientowane.

Multigraf skierowany

Twierdzenie 3

Pętla zorientowaną nazywamy zorientowaną krawędź e, oraz wierzchołek v takie, że e prowadzi z v do v.

Digraf

Twierdzenie 4

Drogą T długości n nazywamy ciąg zgodnie zorientowanych krawędzi $e_1 e_2 \ldots e_n$, oraz wierzchołków $v_1 v_2 \ldots v_{n+1}$, takich że $e_i = (v_i, v_{i+1})$.
W drodze krawędzie mogą się powtarzać.
W drodze wierzchołki mogą się powtarzać.

Digraf

Drogą zamkniętą długości n nazywamy ciąg zgodnie zorientowanych krawędzi $e_1 e_2 \ldots e_n$, oraz wierzchołków $v_1 v_2 \ldots v_{n+1}$, takich że $v_1 = v_{n+1}$.
W drodze zamkniętej krawędzie mogą się powtarzać.
W drodze zamkniętej wierzchołki mogą się powtarzać.

Digraf

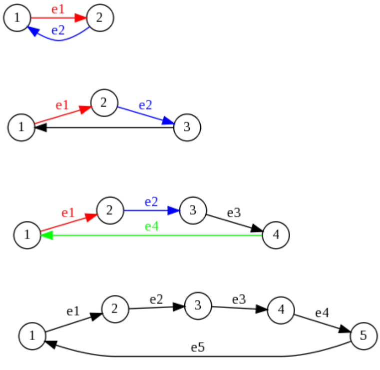
Drogą prostą P długości n nazywamy ciąg różnych zgodnie zorientowanych krawędzi $e_1 e_2 \ldots e_n$, oraz wierzchołków $v_1 v_2 \ldots v_{n+1}$, takich że $e_i = \{v_i, v_{i+1}\}$ (tu uporządkowany zbiór, zatem lepszy zapis w postaci pary uporządkowanej $e_i = (v_i, v_{i+1})$).
W drodze prostej krawędzie nie mogą się powtarzać.
W drodze prostej wierzchołki mogą się powtarzać.

Digraf

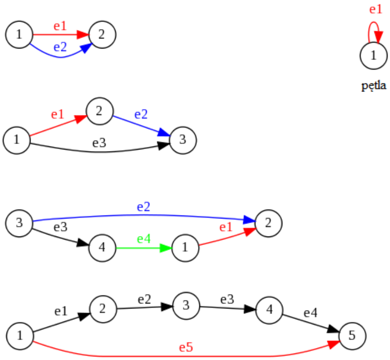
Ścieżką prostą S długości n nazywamy ciąg różnych zgodnie zorientowanych krawędzi $e_1 e_2 \ldots e_n$, oraz ciąg różnych wierzchołków $v_1 v_2 \ldots v_{n+1}$, takich że
$e_i = \{v_i, v_{i+1}\}$
(tu para uporządkowana $e_i = (v_i, v_{i+1})$).
W ścieżce prostej krawędzie nie mogą się powtarzać.
W ścieżce prostej wierzchołki nie mogą się powtarzać.

Digraf

Cyklem C długości n nazywamy ciąg różnych zgodnie zorientowanych krawędzi $e_1 e_2 \dots e_n$, oraz ciąg różnych wierzchołków $v_1 v_2 \dots v_{n+1}$, takich że $v_1 = v_{n+1}$. Przy czym zakładamy, że $n \geq 2$.
W cyklu krawędzie nie mogą się powtarzać.
W cyklu wierzchołki $v_1, v_2, \dots, v_n$ nie mogą się powtarzać.



cykle



nie cykle

Twierdzenie 5

(Droga acykliczna)

Niech G multigraf skierowany (dopuszczamy pętle i krawędzie wielokrotne).
Niech u i v to różne wierzchołki G.
Jeżeli istnieje w G droga z u do v,
to istnieje w G droga acykliczna z u do v.

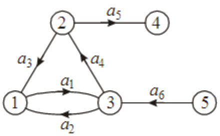
Digraf

Twierdzenie 6

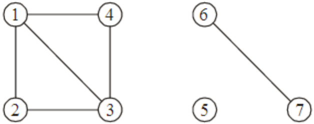
Niech G multigraf skierowany (dopuszczamy pętle i krawędzie wielokrotne).
Niech u to wierzchołek w G
Jeżeli istnieje w G droga zamknięta z u do u,
to istnieje w G cykl z u do u lub krawędź prowadząca z u do u (pętla).

Graf skierowany jest ***spójny*** (*słabo spójny*), jeśli jego pochodny graf nieskierowny jest spójny.
Graf skierowany jest ***silnie spójny***, jeśli dla każdej pary wierzchołków u i v istnieje w nim droga z u do v.
Składową spójną grafu nazywamy jego podgraf, który jest spójny i nie jest podgrafem innego grafu spójnego.

Przykład skierowanego grafu spójnego, ale nie silnie spójnego



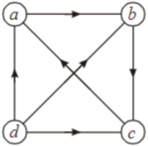
Przykład grafu o 3 składowych spójnych



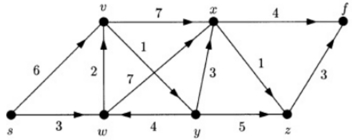
Graf skierowany bez pętli nazywamy ***turniejem***, jeśli dla każdej pary wierzchołków u i v zawiera on dokładnie jeden łuk: albo (u, v), albo (v, u).

- turniej może reprezentować wyniki spotkań par uczestniczących w rozgrywkach typu „każdy z każdym" (bez remisów)

Przykład turnieju



Uwaga: ten turniej nie jest silnie spójny (nie ma np. drogi z b do d)



Jeśli chcemy, możemy przedstawić funkcję wagi W w postaci tablicy, oznaczając wiersze i kolumny tablicy za pomocą elementów zbioru $V(G)$ i pisząc wartość $W(u, v)$ na przecięciu wiersza u z kolumną v.

Liczby znajdują się w miejscach odpowiadających krawędziom tego grafu.

W	s	v	w	x	y	z	f
s		6	3				
v				7	1		
w		2		7			
x						1	4
y			4	3		5	
z							3
f							

Na rysunku mamy tablicę funkcji wagi W^*,
gdzie $W^*(u, v)$ oznacza najmniejszą wagę drogi prowadzącej z wierzchołka u do wierzchołka v, o ile taka droga istnieje.

W^*	s	v	w	x	y	z	f
s		5	3	9	6	10	13
v		7	5	4	1	5	8
w		2	7	6	3	7	10
x						1	4
y		6	4	3	7	4	7
z							3
f							

Wygodnie jest przyjąć oznaczenie $W(u, v) = \infty$, jeśli nie ma krawędzi z u do v w grafie G; symbol $W^*(u, v) = \infty$ oznacza, że w grafie G nie ma drogi z wierzchołka u do wierzchołka v. Mamy następujące reguły wykonywania działań na symbolu ∞: $\infty + x = x + \infty = \infty$ dla każdego x oraz $a < \infty$ dla każdej liczby rzeczywistej a.

Algorytm Dijkstry

{Dane: graf skierowany bez pętli i krawędzi wielokrotnych, którego zbiorem wierzchołków jest $\{1, \ldots, n\}$, funkcja W wag krawędzi o wartościach nieujemnych}

{Wyniki: wagi minimalne $W^*(1, j)$ dla $j = 2, \ldots, n$}

{Zmienne pomocnicze: zbiory L, V oraz funkcja D}

Niech $L := \emptyset$ oraz $V := \{2, \ldots, n\}$.

Dla $i \in V$ wykonuj
  $D(i) := W(1, i)$.
Dopóki $V \setminus L \neq \emptyset$, wykonuj
  wybierz $k \in V \setminus L$ o najmniejszej wartości $D(k)$
  dołącz k do zbioru L
  dla każdego $j \in V \setminus L$ wykonuj
    jeśli $D(j) > D(k) + W(k, j)$, to
      zastąp $D(j)$ sumą $D(k) + W(k, j)$.
Dla $j \in V$ wykonuj
  $W^*(1, j) := D(j)$.

Algorytm Dijkstry

{Dane: graf skierowany bez pętli i krawędzi wielokrotnych, którego zbiorem wierzchołków jest $\{1, \ldots, n\}$, funkcja W wag krawędzi o wartościach nieujemnych}

{Wyniki: wagi minimalne $W^*(1,j)$ dla $j = 2, \ldots, n$}

{Zmienne pomocnicze: zbiory L, V oraz funkcja D}

Niech $L := \emptyset$ oraz $V := \{2, \ldots, n\}$.

Dla $i \in V$ wykonuj

  $D(i) := W(1,i)$.

Dopóki $V \setminus L \neq \emptyset$, wykonuj

  wybierz $k \in V \setminus L$ o najmniejszej wartości $D(k)$

  dołącz k do zbioru L

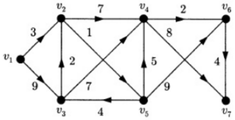
  dla każdego $j \in V \setminus L$ wykonuj

    jeśli $D(j) > D(k) + W(k,j)$, to

      zastąp $D(j)$ sumą $D(k) + W(k,j)$.

Dla $j \in V$ wykonuj

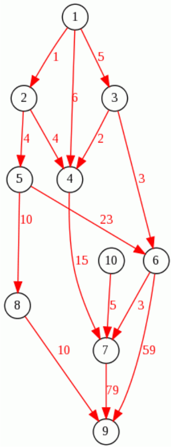
  $W^*(1,j) := D(j)$.



W	v_1	v_2	v_3	v_4	v_5	v_6	v_7
v_1	∞	3	9	∞	∞	∞	∞
v_2	∞	∞	∞	7	1	∞	∞
v_3	∞	2	∞	7	∞	∞	∞
v_4	∞	∞	∞	∞	∞	2	8
v_5	∞	∞	4	5	∞	9	∞
v_6	∞	∞	∞	∞	∞	∞	4
v_7	∞	∞	∞	∞	∞	∞	∞

L	$D(2)$	$D(3)$	$D(4)$	$D(5)$	$D(6)$	$D(7)$	Komentarz	Obliczenia
$\emptyset$	3	9	∞	∞	∞	∞	dane początkowe	
$\{2\}$	3	9	10	4	∞	∞	znaleziono drogi $v_1v_2v_4$ oraz $v_1v_2v_5$	
$\{2,5\}$	3	8	9	4	13	∞	znaleziono drogi $v_1v_2v_5v_3$, $v_1v_2v_5v_4$ oraz $v_1v_2v_5v_6$	jeśli $D(j) > D(k) + W(k,j)$, to zastąp $D(j)$ sumą $D(k) + W(k,j)$
$\{2,5,3\}$	3	8	9	4	13	∞	brak poprawy	
$\{2,5,3,4\}$	3	8	9	4	11	17	znaleziono drogi $v_1v_2v_5v_4v_6$ oraz $v_1v_2v_5v_4v_7$	
$\{2,5,3,4,6\}$	3	8	9	4	11	15	znaleziono drogi $v_1v_2v_5v_4v_6v_7$	

Algorytm Dijkstry



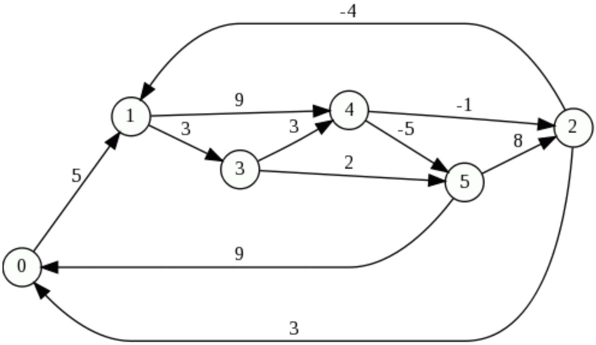
Algorytm Bellmana-Forda

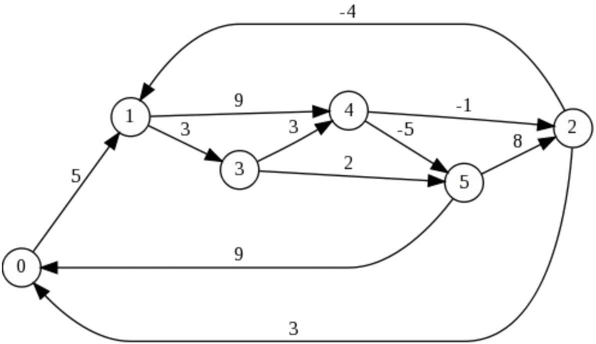
W silnie spójnym grafie ważonym (dopuszamy ujemne wagi) znaleźć najkrótsze ścieżki proste od wybranego startowego 0 **wierzchołka do wszystkich pozostałych wierzchołków.**

Niezbędne założenie graf nie zawiera **ujemnego cyklu, pętli** czyli cyklu, w którym suma wag krawędzi jest ujemna lub pętli o ujemnej wadze.

Tworzymy dwie n elementowe tablice danych, $n = |V(G)|$.

$D(i)$ to waga najkrótszej ścieżki prostej z wierzchołka startowego do $i-$tego wierzchołka grafu. $D(0) = 0$.
$P(i)$ to numer wierzchołka grafu, który jest poprzednikiem wierzchołka na najkrótszej ścieżce. $P(0) = -1$.

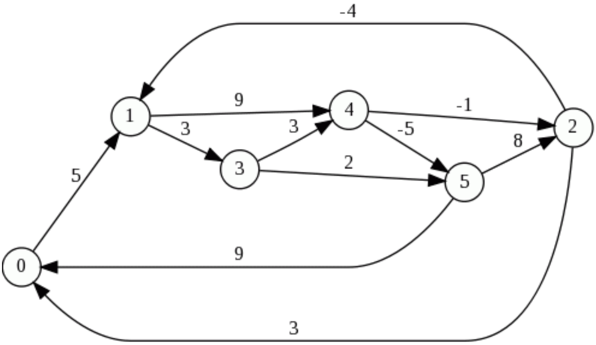

-4
9
-1
3
3
-5
8
2
5
9
3
0
1
2
3
4
5



i	0	1	2	3	4	5
D[i]	0	∞	∞	∞	∞	∞
P[i]	-1	-1	-1	-1	-1	-1

Wybieramy na wierzchołek startowy wierzchołek o numerze zero.

Tworzymy tablice d i p, wypełniając je odpowiednio



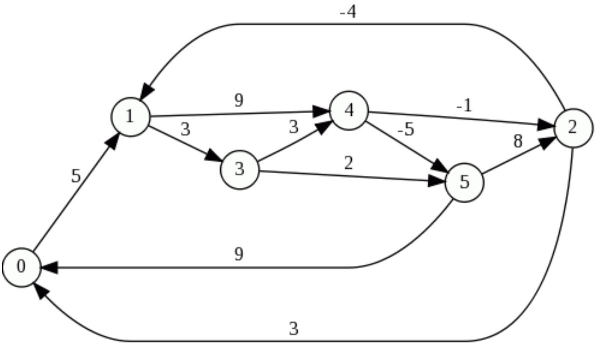
W pierwszym kroku mamy:

D[0]=0
D[1]=∞
D[1]>D[0]+5

Ustawiamy:

D[1]=0+5=5
P[1]=0

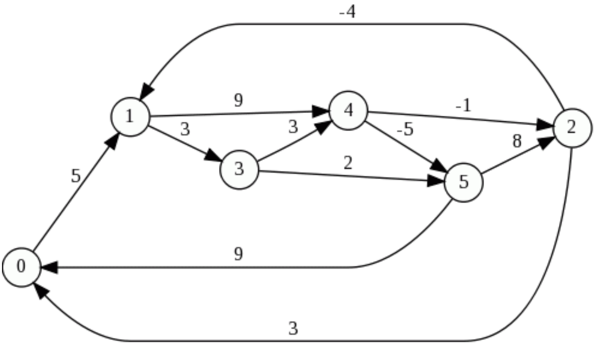
i	0	1	2	3	4	5
D[*i*]	0	5	∞	∞	∞	∞
P[*i*]	-1	0	-1	-1	-1	-1



W drugim kroku mamy:

D[1]=5 D[3]= ∞ D[3]>D[1]+3 Ustawiamy: D[3]=5+3=8 P[3]=1	D[1]=5 D[4]= ∞ D[4]>D[1]+9 Ustawiamy: D[4]=5+9=14 P[4]=1

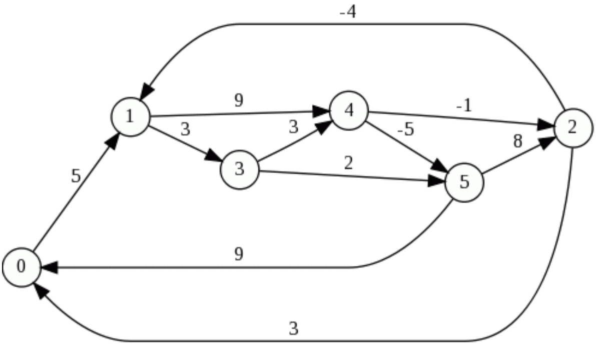
i	0	1	2	3	4	5
D[*i*]	0	5	∞	8	14	∞
P[*i*]	-1	0	-1	1	1	-1



W trzecim kroku:

D[4]=14 D[2]= ∞ D[2]>D[4]+(-1) Ustawiamy: D[2]=14+(-1)=13 P[2]=4	D[4]=14 D[5]= ∞ D[5]>D[4]+(-5) Ustawiamy: D[5]=14+(-5)=9 D[5]=4	D[3]=8 D[4]=14 D[4]>D[3]+3 Ustawiamy: D[4]=8+3=11 D[4]=3

i	0	1	2	3	4	5
D[*i*]	0	5	13	8	11	9
P[*i*]	-1	0	4	1	3	4



W czwartym kroku

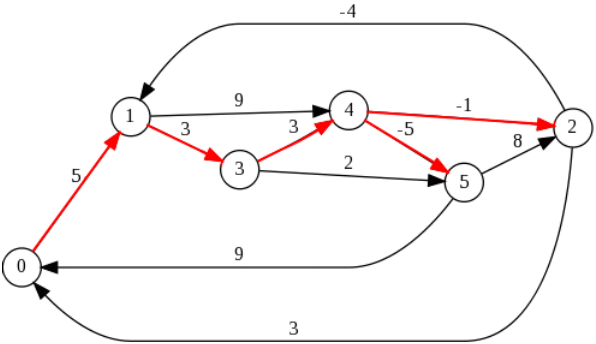
D[4]=11
D[2]=13
D[2]>D[4]+(-1)

Ustawiamy:
D[2]=11+(-1)=10
P[2]=4

D[4]=11
D[5]=9
D[5]>D[4]+(-5)

Ustawiamy:
D[5]=11+(-5)=6
P[5]=4

i	0	1	2	3	4	5
D[*i*]	0	5	10	8	11	6
P[*i*]	-1	0	4	1	3	4



Krok piąty nie prowadzi do zmian.
Odpowiednia nierówność nie będzie spełniona dla żadnej krawędzi.

Tablica D zawiera minimalne koszty dojścia od wierzchołka 0 do poszczególnych wierzchołków w grafie.

Tablica P zawiera informacje o najkrótszych ścieżkach prostych od wierzchołka 0 do poszczególnych wierzchołków w grafie.

i	0	1	2	3	4	5
D[*i*]	0	5	10	8	11	6
P[*i*]	-1	0	4	1	3	4

Sortowanie topologiczne (ang. topological sort) grafu skierowanego polega na utworzeniu listy wierzchołków grafu w taki sposób, aby każdy wierzchołek posiadający sąsiadów znalazł się na tej liście przed nimi.
skierowanym (ang. DAG - directed acyclic graph). Kolejność wierzchołków nie połączonych ścieżką jest dowolna.

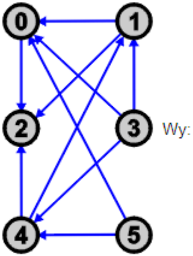
Zadanie to jest możliwe do wykonania tylko wtedy, gdy graf jest **acyklicznym grafem**

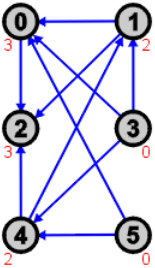
Do sortowania topologicznego grafu wykorzystamy własność, która mówi, że w acyklicznym grafie skierowanym musi wystąpić przynajmniej jeden wierzchołek o stopniu wchodzącym zero (czyli wierzchołek, który nie jest sąsiadem żadnego innego wierzchołka w grafie). Usunięcie wierzchołka z grafu nigdy nie powoduje, że staje się on grafem cyklicznym.

Zatem po każdej operacji usunięcia wierzchołka w grafie pojawia się przynajmniej jeden wierzchołek o stopniu wchodzącym zero.

Z grafu usuwamy wierzchołki o stopniu wchodzącym zero dotąd, aż będzie on pusty lub nie znajdziemy wierzchołka o stopniu wchodzącym zero, co oznacza, że graf jest cykliczny i nie może być posortowany topologicznie.

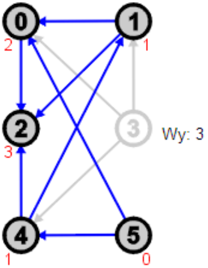
Usuwane wierzchołki tworzą listę wierzchołków grafu posortowanego topologicznie.


0
1
2
3
4
5
Wy:



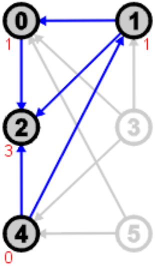
Wy:

Wyznaczamy stopnie wchodzące poszczególnych wierzchołków.
Dwa wierzchołki mają stopień wchodzący równy zero: wierzchołek 3 i 5.
Możemy usunąć dowolny z nich, ponieważ nie są one od siebie zależne.



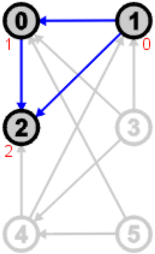
Usuwamy z grafu wierzchołek nr 3 wraz z krawędziami i przesyłamy go na wyjście. Powoduje to zmianę stopni wchodzących wierzchołków sąsiednich.

W grafie wciąż pozostał wierzchołek 5 o stopniu wejściowym zero.



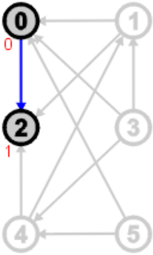
Wy: 3 5

Usuwamy z grafu wierzchołek 5 i przesyłamy go na wyjście.
W grafie zmieniły się stopnie wchodzące wierzchołków sąsiednich.
Teraz wierzchołek 4 ma stopień wchodzący równy zero.



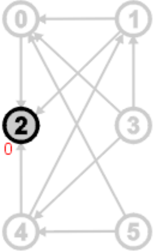
Wy: 3 5 4

Usuwamy z grafu wierzchołek 4 i przesyłamy go na wyjście.
W wyniku usunięcia wierzchołka 4 wierzchołek 1 ma stopień wchodzący zero.



Wy: 3 5 4 1

Usuwamy z grafu wierzchołek 1 i przesyłamy go na wyjście.
Stopień wchodzący zero otrzymuje wierzchołek 0.



Wy: 3 5 4 1 0

Usuwamy z grafu wierzchołek 0 i przesyłamy go na wyjście.
W grafie pozostał tylko wierzchołek 2 i ma on teraz stopień wchodzący 0.

Wy: **3 5 4 1 0 2** Usuwamy z grafu ostatni wierzchołek 2 i przesyłamy go na wyjście. Graf stał się grafem pustym i nie posiada już dalszych wierzchołków. Sortowanie topologiczne jest zakończone.