Teoria grafów - część II

Aleksandra Gorzkowska

Elektronika i Telekomunikacja
Wydział Informatyki, Elektroniki i Telekomunikacji

Definicja

Drogą w grafie $G = (V, E)$ nazywamy ciąg $v_0 e_1 v_1 e_2 v_2 \dots v_{k-1} e_k v_k$, gdzie $v_{i-1} v_i = e_i \in E$ dla każdego $i \in \{1, 2, \dots, k\}$.

Jest to ciąg naprzemienny wierzchołków i krawędzi, taki że dwa kolejne wierzchołki są oddzielone krawędzią, którą tworzą w grafie.

Definicja

Drogą w grafie $G = (V, E)$ nazywamy ciąg $v_0 e_1 v_1 e_2 v_2 \dots v_{k-1} e_k v_k$, gdzie $v_{i-1} v_i = e_i \in E$ dla każdego $i \in \{1, 2, \dots, k\}$.

Jest to ciąg naprzemienny wierzchołków i krawędzi, taki że dwa kolejne wierzchołki są oddzielone krawędzią, którą tworzą w grafie.

W skrócie zapisujemy $v_0 v_1 v_2 \dots v_k$.

Definicja

Drogą w grafie $G = (V, E)$ nazywamy ciąg $v_0 e_1 v_1 e_2 v_2 \dots v_{k-1} e_k v_k$, gdzie $v_{i-1} v_i = e_i \in E$ dla każdego $i \in \{1, 2, \dots, k\}$.

Jest to ciąg naprzemienny wierzchołków i krawędzi, taki że dwa kolejne wierzchołki są oddzielone krawędzią, którą tworzą w grafie.

W skrócie zapisujemy $v_0 v_1 v_2 \dots v_k$.

Liczbę krawędzi drogi, czyli liczbę k, nazywamy **długością** drogi.

Definicja

Drogą w grafie $G = (V, E)$ nazywamy ciąg $v_0 e_1 v_1 e_2 v_2 \ldots v_{k-1} e_k v_k$, gdzie $v_{i-1} v_i = e_i \in E$ dla każdego $i \in \{1, 2, \ldots, k\}$.

Jest to ciąg naprzemienny wierzchołków i krawędzi, taki że dwa kolejne wierzchołki są oddzielone krawędzią, którą tworzą w grafie.

W skrócie zapisujemy $v_0 v_1 v_2 \ldots v_k$.

Liczbę krawędzi drogi, czyli liczbę k, nazywamy długością drogi.

- Jeśli $v_0 = v_k$, to drogę nazywamy zamkniętą.

Definicja

Drogą w grafie $G = (V, E)$ nazywamy ciąg $v_0 e_1 v_1 e_2 v_2 \ldots v_{k-1} e_k v_k$, gdzie $v_{i-1} v_i = e_i \in E$ dla każdego $i \in \{1, 2, \ldots, k\}$.

Jest to ciąg naprzemienny wierzchołków i krawędzi, taki że dwa kolejne wierzchołki są oddzielone krawędzią, którą tworzą w grafie.

W skrócie zapisujemy $v_0 v_1 v_2 \ldots v_k$.

Liczbę krawędzi drogi, czyli liczbę k, nazywamy długością drogi.

- Jeśli $v_0 = v_k$, to drogę nazywamy zamkniętą.
- Jeśli $v_i \neq v_j$, $\forall i \neq j$, to drogę nazywamy ścieżką.

Definicja

Drogą w grafie $G = (V, E)$ nazywamy ciąg $v_0 e_1 v_1 e_2 v_2 \dots v_{k-1} e_k v_k$, gdzie $v_{i-1} v_i = e_i \in E$ dla każdego $i \in \{1, 2, \dots, k\}$.

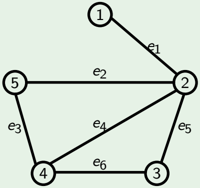
Jest to ciąg naprzemienny wierzchołków i krawędzi, taki że dwa kolejne wierzchołki są oddzielone krawędzią, którą tworzą w grafie.

W skrócie zapisujemy $v_0 v_1 v_2 \dots v_k$.

Liczbę krawędzi drogi, czyli liczbę k, nazywamy długością drogi.

- Jeśli $v_0 = v_k$, to drogę nazywamy zamkniętą.
- Jeśli $v_i \neq v_j$, $\forall i \neq j$, to drogę nazywamy ścieżką.
- Jeśli w drodze zamkniętej $(v_0 = v_k)$ $v_i \neq v_j$, $i, j \in \{1, 2, \dots, k\}$, to drogę nazywamy cyklem.

Przykład



graf G

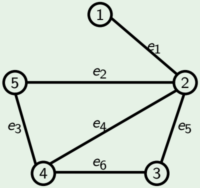
$2e_4 4e_3 5e_2 2e_5 3$ - droga

$2e_4 4e_6 3e_5 2e_2 5e_3 4e_4 2$ - droga zamknięta

$1e_1 2e_5 3e_6 4$ - ścieżka

$2e_2 5e_3 4e_6 3e_5 2$ - cykl

Przykład



graf G

$2e_44e_35e_22e_53$ - droga

$2e_44e_63e_52e_25e_34e_42$ - droga zamknięta

$1e_12e_53e_64$ - ścieżka

$2e_25e_34e_63e_52$ - cykl

Jeśli w grafie G istnieje droga/ścieżka $v_0v_1 \ldots v_k$, to v_0 oraz v_k nazywamy końcami tej drogi/ścieżki. Mówimy również, że ta droga/ścieżka łączy wierzchołki v_0 oraz v_k.

Definicja

Graf G nazywamy spójnym, jeśli dla dowolnych dwóch wierzchołków $x, y \in V(G)$ istnieje ścieżka łącząca je w grafie G.

Definicja

Graf G nazywamy spójnym, jeśli dla dowolnych dwóch wierzchołków $x, y \in V(G)$ istnieje ścieżka łącząca je w grafie G.

Definicja

Maksymalny (w sensie inkluzji) spójny podgraf grafu G nazywamy składową grafu G.

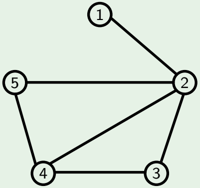
Definicja

Graf G nazywamy spójnym, jeśli dla dowolnych dwóch wierzchołków $x, y \in V(G)$ istnieje ścieżka łącząca je w grafie G.

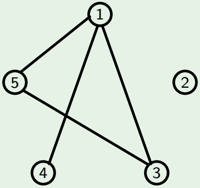
Definicja

Maksymalny (w sensie inkluzji) spójny podgraf grafu G nazywamy składową grafu G.

Przykład



graf G jest spójny



graf $\overline{G}$ NIE jest spójny

Dwie składowe: wierzchołek 2; podgraf indukowany (grafu $\overline{G}$) przez zbiór wierzchołków $\{1, 3, 4, 5\}$

Definicja

Mostem w grafie spójnym G nazywamy krawędź, po usunięciu której graf przestaje być spójny.

Definicja

Mostem w grafie spójnym G nazywamy krawędź, po usunięciu której graf przestaje być spójny.

Definicja

Wierzchołkiem rozspajającym w grafie spójnym G nazywamy wierzchołek, po usunięciu którego graf przestaje być spójny.

Definicja

Drogę zamkniętą, w której każda krawędź grafu G występuje dokładnie raz nazywamy obchodem Eulera (potocznie zwanym cyklem Eulera).

Graf, w którym istnieje obchód Eulera nazywamy eulerowskim.

Definicja

Drogę zamkniętą, w której każda krawędź grafu G występuje dokładnie raz nazywamy obchodem Eulera (potocznie zwanym cyklem Eulera).

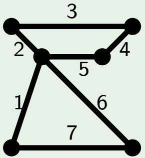
Graf, w którym istnieje obchód Eulera nazywamy eulerowskim.

Definicja

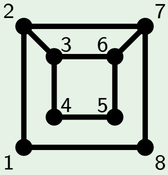
Cykl zawierający każdy wierzchołek grafu G nazywamy cyklem Hamiltona.

Graf, w którym istnieje cykl Hamiltona nazywamy hamiltonowskim.

Przykład



graf eulerowski



graf hamiltonowski

Numeracja wskazuje kolejność krawędzi/wierzchołków na drodze zamkniętej.

Warunki konieczne hamiltonowskości

Twierdzenie

Jeśli graf jest hamiltonowski, to nie istnieje w nim wierzchołek rozspajający.

Warunki konieczne hamiltonowskości

Twierdzenie

Jeśli graf jest hamiltonowski, to nie istnieje w nim wierzchołek rozspajający.

Uwaga! Stwierdzenie odwrotne nie jest prawdziwe.

Warunki konieczne hamiltonowskości

Twierdzenie

Jeśli graf jest hamiltonowski, to nie istnieje w nim wierzchołek rozspajający.

Uwaga! Stwierdzenie odwrotne nie jest prawdziwe.

Twierdzenie

Jeśli graf G jest hamiltonowski, $S \subset V(G)$, to graf powstały po usunięciu zbioru S z grafu G ma co najwyżej $|S|$ składowych.

Warunki konieczne hamiltonowskości

Twierdzenie

Jeśli graf jest hamiltonowski, to nie istnieje w nim wierzchołek rozspajający.

Uwaga! Stwierdzenie odwrotne nie jest prawdziwe.

Twierdzenie

Jeśli graf G jest hamiltonowski, $S \subset V(G)$, to graf powstały po usunięciu zbioru S z grafu G ma co najwyżej $|S|$ składowych.

Uwaga! Stwierdzenie odwrotne nie jest prawdziwe.

Warunki wystarczające hamiltonowskości

Twierdzenie

Niech $G = (V, E)$ będzie grafem prostym rzędu n. Jeśli $u, v \in V$: $uv \notin E$, $d(u) + d(v) \geq n$ oraz $G + uv$ jest hamiltonowski, to G jest hamiltonowski.

Warunki wystarczające hamiltonowskości

Twierdzenie

Niech $G = (V, E)$ będzie grafem prostym rzedu n. Jeśli $u, v \in V : uv \notin E$, $d(u) + d(v) \geq n$ oraz $G + uv$ jest hamiltonowski, to G jest hamiltonowski.

Definicja

k-tym domknięciem Bondy'ego-Chvátala grafu G nazywamy graf powstały z G poprzez iteracyjne dodawanie krawędzi między niepołączonymi wierzchołkami, których suma stopni wynosi co najmniej k.
Oznaczenie: $cl_k(G)$

Warunki wystarczające hamiltonowskości

Twierdzenie

Niech $G = (V, E)$ *będzie grafem prostym rzedu* n. *Jeśli* $u, v \in V : uv \notin E$, $d(u) + d(v) \geq n$ *oraz* $G + uv$ *jest hamiltonowski, to* G *jest hamiltonowski.*

Twierdzenie

Jeśli G *jest grafem rzędu* n *oraz* $cl_n(G)$ *jest hamiltonowski (w szczególności* $cl_n(G) = K_n$*), to* G *jest hamiltonowski.*

Warunki wystarczające hamiltonowskości

Twierdzenie (Orego)

Jeśli $G = (V, E)$ *jest grafem rzędu* n *oraz*
$\forall u, v \in V : uv \notin E \Rightarrow d(u) + d(v) \geq n$, *to* G *jest hamiltonowski.*

Warunki wystarczające hamiltonowskości

Twierdzenie (Orego)

Jeśli $G = (V, E)$ *jest grafem rzędu* n *oraz*
$\forall u, v \in V : uv \notin E \Rightarrow d(u) + d(v) \geq n$, *to* G *jest hamiltonowski.*

Twierdzenie (Diracka)

Jeśli G *jest grafem rzędu* n *oraz* $\delta(G) \geq \frac{n}{2}$, *to* G *jest hamiltonowski*

Warunki wystarczające hamiltonowskości

Założenia tych twierdzeń nazywamy warunkami Orego i Diracka.

Warunki wystarczające hamiltonowskości

Założenia tych twierdzeń nazywamy warunkami Orego i Diracka.

Niech $G = (V, E)$ będzie grafem rzedu n. Mówimy, że G spełnia:

Warunki wystarczające hamiltonowskości

Założenia tych twierdzeń nazywamy warunkami Orego i Diracka.

Niech $G = (V, E)$ będzie grafem rzedu n. Mówimy, że G spełnia:

warunek Orego, jeśli: $\forall u, v \in V : uv \notin E \Rightarrow d(u) + d(v) \geq n$

Warunki wystarczające hamiltonowskości

Założenia tych twierdzeń nazywamy warunkami Orego i Diracka.

Niech $G = (V, E)$ będzie grafem rzedu n. Mówimy, że G spełnia:

warunek Orego, jeśli: $\forall u, v \in V : uv \notin E \Rightarrow d(u) + d(v) \geq n$

warunek Diracka, jesli: $\delta(G) \geq \frac{n}{2}$

Warunki wystarczające hamiltonowskości

Założenia tych twierdzeń nazywamy warunkami Orego i Diracka.

Niech $G = (V, E)$ będzie grafem rzedu n. Mówimy, że G spełnia:

warunek Orego, jeśli: $\forall u, v \in V : uv \notin E \Rightarrow d(u) + d(v) \geq n$

warunek Diracka, jesli: $\delta(G) \geq \frac{n}{2}$

Każdy graf, który spełnia warunek Diracka spełnia również warunek Orego.

Warunki wystarczające hamiltonowskości

Założenia tych twierdzeń nazywamy warunkami Orego i Diracka.

Niech $G = (V, E)$ będzie grafem rzedu n. Mówimy, że G spełnia:

warunek Orego, jeśli: $\forall u, v \in V : uv \notin E \Rightarrow d(u) + d(v) \geq n$

warunek Diracka, jesli: $\delta(G) \geq \frac{n}{2}$

Każdy graf, który spełnia warunek Diracka spełnia również warunek Orego.

Nie jest prawdą, że każdy graf, który spełnia warunek Orego spełnia również warunek Diracka.

Definicja

Graf spójny bez cykli nazywamy drzewem.

Definicja

Graf spójny bez cykli nazywamy drzewem.

Graf, którego składowe są drzewami nazywamy lasem.

Definicja

Graf spójny bez cykli nazywamy drzewem.

Graf, którego składowe są drzewami nazywamy lasem.

Twierdzenie

Niech T będzie grafem rzędu n. Następujące warunki są równoważne:

1. *T jest drzewem,*
2. *dowolne dwa wierzchołki T łączy dokładnie jedna ścieżka,*
3. *T jest spójny i usunięcie dowolnej krawędzi powoduje rozspójnienie T,*
4. *T jest spójny i ma $n-1$ krawędzi.*

Definicja

Graf spójny bez cykli nazywamy drzewem.

Graf, którego składowe są drzewami nazywamy lasem.

Twierdzenie

Niech T będzie grafem rzędu n. Następujące warunki są równoważne:

1. *T jest drzewem,*
2. *dowolne dwa wierzchołki T łączy dokładnie jedna ścieżka,*
3. *T jest spójny i usunięcie dowolnej krawędzi powoduje rozspójnienie T,*
4. *T jest spójny i ma $n-1$ krawędzi.*

Oznacza to, że dowolny z powyższych warunków może zostać użyty jako definicja drzewa.

Twierdzenie

Każde drzewo rzędu co najmniej dwa ma co najmniej jeden liść.

Twierdzenie

Każde drzewo rzędu co najmniej dwa ma co najmniej jeden liść.

Dowód.

Załóżmy dla dowodu nie wprost, że w drzewie T nie ma liści.
Ponieważ jest to graf spójny, to każdy wierzchołek musi mieć stopień co najmniej 2.

$$\forall v \in V(T) \ \ \mathrm{d}(v) \geq 2$$

Z poprzedniego twierdzenia wiemy, że jeśli T jest drzewem rzędu n, to ma rozmiar równy $n-1$.
Skorzystamy z lematu o uściskach dłoni.

$$2(n-1) = \sum_{v \in V(T)} \mathrm{d}(v) \geq 2 \cdot n$$

$2n - 2 \geq 2n$ - sprzeczność! $\square$

Definicja

Drzewem ukorzenionym nazywamy drzewo z wyróżnionym jednym (dowolnym) wierzchołkiem. Wierzchołek ten nazywamy korzeniem.

Definicja

Drzewem ukorzenionym nazywamy drzewo z wyróżnionym jednym (dowolnym) wierzchołkiem. Wierzchołek ten nazywamy korzeniem.

Definicja

Drzewem binarnym nazywamy drzewo ukorzenione, którego korzeń ma stopień 0 lub 2, a każdy inny wierzchołek ma stopień 1 lub 3.

Definicja

Drzewem ukorzenionym nazywamy drzewo z wyróżnionym jednym (dowolnym) wierzchołkiem. Wierzchołek ten nazywamy korzeniem.

Definicja

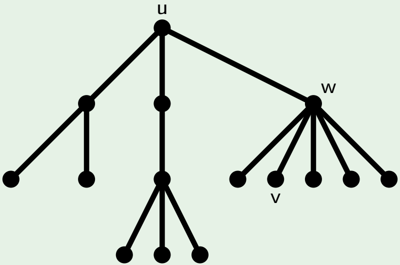
Drzewem binarnym nazywamy drzewo ukorzenione, którego korzeń ma stopień 0 lub 2, a każdy inny wierzchołek ma stopień 1 lub 3.

Definicja

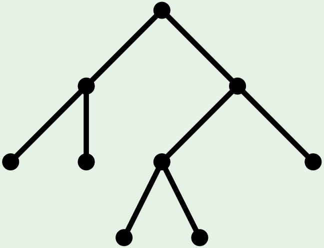
W drzewie ukorzenionym o korzeniu u:

- ojcem wierzchołka v nazywamy wierzchołek bezpośrednio przed v na ścieżce łączącej u z v,
- synem wierzchołka v nazywamy każdy wierzchołek w, taki że v jest bezpośrednio przed w na ścieżce łączącej u z w.

Przykład



drzewo ukorzenione; korzeń u



drzewo binarne

Definicja

Drzewem rozpinającym grafu $G = (V, E)$ nazywamy dowolny jego podgraf $T = (V, F)$, który jest drzewem.

Definicja

Drzewem rozpinającym grafu $G = (V, E)$ nazywamy dowolny jego podgraf $T = (V, F)$, który jest drzewem.

Jeśli graf ma drzewo rozpinające, to jest spójny. Stwierdzenie odwrotne również jest prawdziwe. W każdym grafie spójnym istnieje drzewo rozpinające.

Definicja

Drzewem rozpinającym grafu $G = (V, E)$ nazywamy dowolny jego podgraf $T = (V, F)$, który jest drzewem.

Jeśli graf ma drzewo rozpinające, to jest spójny. Stwierdzenie odwrotne również jest prawdziwe. W każdym grafie spójnym istnieje drzewo rozpinające.

Drzewo rozpinające grafu G ma rząd $|G|$ i rozmiar $|G| - 1$.

Definicja

Drzewem rozpinającym grafu $G = (V, E)$ nazywamy dowolny jego podgraf $T = (V, F)$, który jest drzewem.

Jeśli graf ma drzewo rozpinające, to jest spójny. Stwierdzenie odwrotne również jest prawdziwe. W każdym grafie spójnym istnieje drzewo rozpinające.

Drzewo rozpinające grafu G ma rząd $|G|$ i rozmiar $|G| - 1$.

W jaki sposób sprawdzić czy graf jest spójny? Jak znaleźć jego drzewo rozpinające?

Definicja

Drzewem rozpinającym grafu $G = (V, E)$ nazywamy dowolny jego podgraf $T = (V, F)$, który jest drzewem.

Jeśli graf ma drzewo rozpinające, to jest spójny. Stwierdzenie odwrotne również jest prawdziwe. W każdym grafie spójnym istnieje drzewo rozpinające.

Drzewo rozpinające grafu G ma rząd $|G|$ i rozmiar $|G| - 1$.

W jaki sposób sprawdzić czy graf jest spójny? Jak znaleźć jego drzewo rozpinające?

Opiszemy dwa algorytmy pozwalające przeszukać wszystkie wierzchołki grafu i znaleźć drzewo rozpinające.

DFS - przeszukiwanie w głąb

Dany graf $G = (V, E)$.

dla każdego wierzchołka $v \in V$ ustaw odwiedzony[v]=0;

DFS(v) {
    odwiedzony[v]=1;
    dla każdego $u \in V$ sąsiada v
        jeśli odwiedzony[u]=0 {
            dodaj vu do drzewa T;
            wykonaj DFS(u);
        }
}

Wywołanie procedury odbywa się na dowolnym wierzchołku grafu.

Jeśli po zakończeniu działania programu wszystkie wierzchołki są odwiedzone, to graf jest spójny oraz otrzymane drzewo T jest drzewem rozpinającym grafu G.

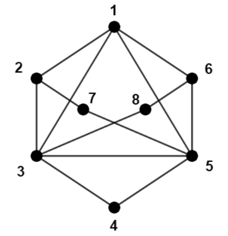
Jeśli istnieje wierzchołek, który nie został odwiedzony, to możemy wykonać na nim kolejny DFS (itd.), aby znaleźć las spinający grafu G. Wtedy graf G jest niespójny.

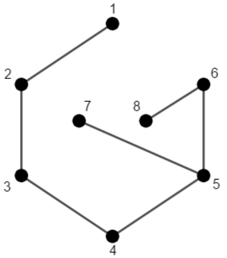
BFS - przeszukiwanie wszerz

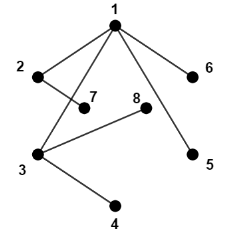
dla każdego wierzchołka $v \in V$ ustaw odwiedzony[v]=0;

BFS(v) {
    kolejka jest pusta;
    odwiedzony[v]=1;
    wstaw v do kolejki;
    dopóki kolejka nie jest pusta {
        pobierz w z kolejki;
        dla każdego $u \in V$ sąsiada w {
            jeśli odwiedzony[u]=0 {
                odwiedzony[u]=1;
                dodaj vu do drzewa T;
                wstaw u do kolejki;
            }
        }
    }
}

Przykład







Graf G wraz z drzewem rozpinającym znalezionym metodą przeszukiwania w głąb i wszerz.

Kod Prüfera

Kod Prüfera pozwala zakodować dowolne drzewo etykietowane rzędu n w postaci ciągu długości $n-2$.

Algorytm kodujący
Dane: drzewo T o zbiorze wierzchołków $\{1, 2, \ldots, n\}$
Szukane: ciąg $(p_1, p_2, \ldots, p_{n_2})$

dla i od 1 do $n-2$ $\{$
    wybierz v_i - liść o najmniejszym indeksie;
    wpisz p_i jako sąsiada wybranego v_i;
    usuń v_i z drzewa T;
    $\}$

Kod Prüfera

Algorytm dekodujący

Dane: ciąg $P = (p_1, p_2, \ldots, p_{n_2})$

Szukane: drzewo T o zbiorze wierzchołków $\{1, 2, \ldots, n\}$

Utwórz listę $L = (1, 2, \ldots, n)$ i las o wierzchołkach izolowanych indeksowanych z listy L

dla i od 1 do $n - 2$ $\{$

wybierz v_i - najmniejszą liczbę z L, która nie występuje w P;

do drzewa T dodaj krawędź $v_i p_i$;

usuń v_i z listy L oraz p_i z ciągu P;

$\}$

Na liście L zostały dwa wierzchołki $\{a, b\}$. Do drzewa T dodaj krawędź ab.