MINIMALNE DRZEWO ROZPINAJĄCE

Dany jest graf G spójny z wagami. Graf nieskierowany jest spójny, jeśli każdy wierzchołek jest osiągalny ze wszystkich innych wierzchołków.

Drzewo rozpinające – podgraf grafu G, który jest drzewem zawierającym wszystkie wierzchołki grafu G.

Minimalne drzewo rozpinające – drzewo rozpinające, dla którego suma wag krawędzi jest minimalna.

Przykład

Przy projektowaniu układów elektronicznych często końcówki wielu elementów składowych należy uczynić elektrycznie równoważnymi, łącząc je przewodami. Do połączenia zbioru n końcówek możemy użyć n-1 przewodów, z których każdy łączy dwie końcówki. Ze wszystkich możliwych sposobów połączeń najbardziej pożądany jest zazwyczaj ten, który minimalizuje łączną długość użytych przewodów.

Problem łączenia końcówek można modelować za pomocą spójnego grafu nieskierowanego z wagami $G=(V,E,w)$, w którym V jest zbiorem końcówek, a E jest zbiorem możliwych połączeń między parami końcówek. Z każdą krawędzią jest związana waga określająca koszt (długość potrzebnego przewodu) połączenia dwóch wierzchołków.

Problem: Znaleźć acykliczny podzbiór krawędzi $T \subset E$, który łączy wszystkie wierzchołki i którego łączna waga jest najmniejsza.

Rozwiązanie: minimalne drzewo rozpinające.

Reguła zachłanna:

- w algorytmie Kruskala – zbiór T jest lasem. Do T jest zawsze dodawana ta krawędź w grafie, która ma najmniejszą wagę i która łączy dwie różne składowe (tzn. dwa drzewa z lasu).
- w algorytmie Prima zbiór T jest zawsze pojedynczym drzewem. Do T jest zawsze dodawana ta krawędź w grafie, która ma najmniejszą wagę i która łączy drzewo wyznaczone przez T z wierzchołkiem spoza tego drzewa.

ALGORYTM KRUSKALA

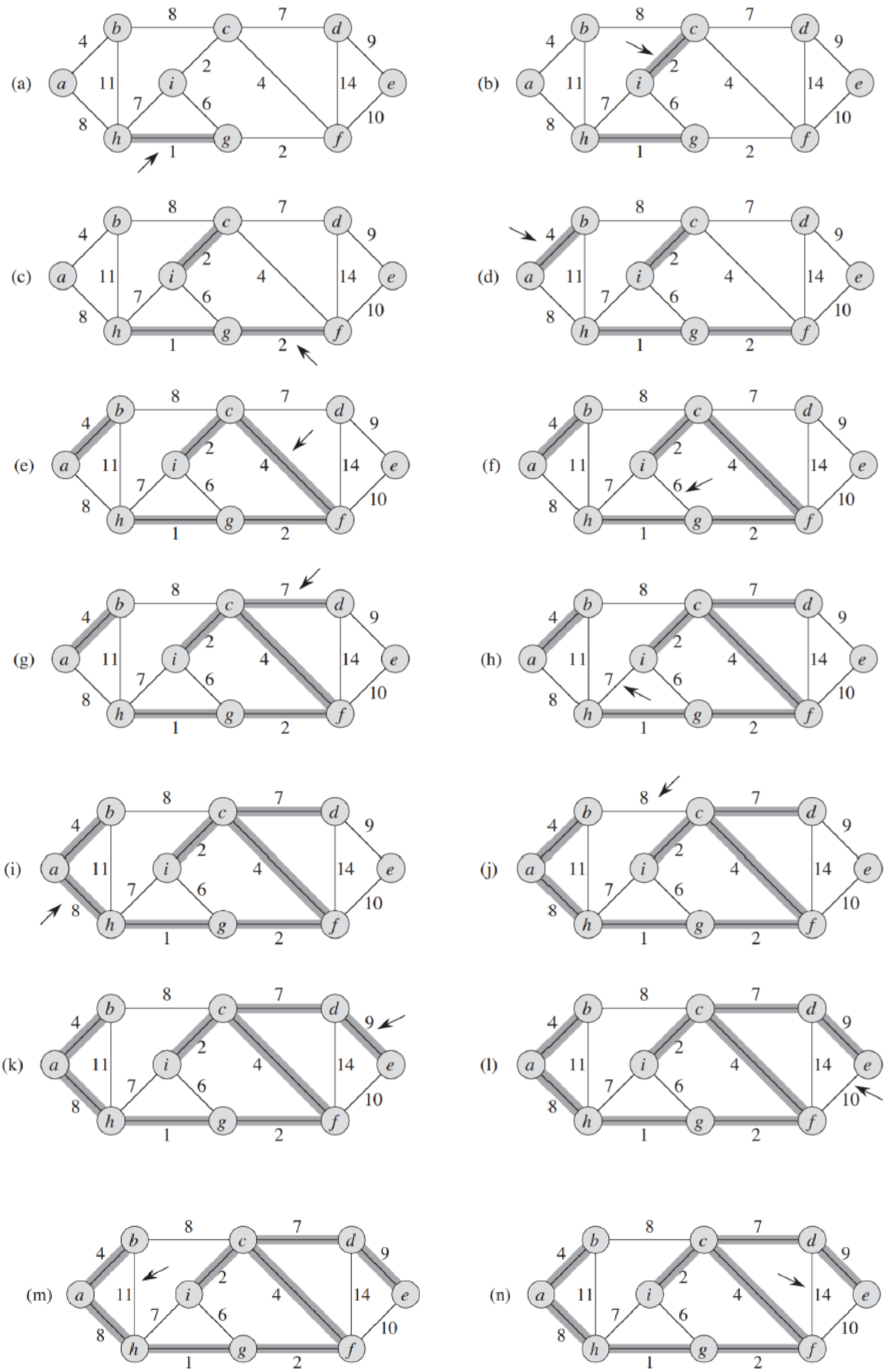
Algorytm Kruskala jest algorytmem zachłannym znajdującym minimalne drzewo rozpinające danego grafu spójnego z wagami. Reguła zachłanna: dodaj krawędź o minimalnej wadze, która nie tworzy cyklu. Rozwiązanie częściowe nie musi być drzewem.

Wejście: $G=(V,E,w)$ – graf spójny nieskierowany z wagami.

Wyjście: T – zbiór krawędzi minimalnego drzewa rozpinającego grafu G.

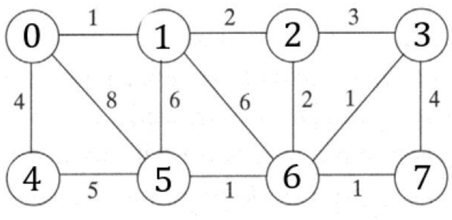
Algorytm:

1. $T=\emptyset$
2. utwórz rozłączne podzbiory zbioru V (każdy podzbiór zawiera jeden wierzchołek ze zbioru V);
3. sortuj zbiór krawędzi E w porządku niemalejącym ze względu na wagi krawędzi;
4. for (każda krawędź (u,v) z uporządkowanego zbioru krawędzi E)
5.   if(u i v należą do podzbiorów rozłączonych)
6.   połącz podzbiory zawierające u i v;
7.   dodaj krawędź (u,v) do zbioru T


(a)
(b)
(c)
(d)
(e)
(f)
(g)
(h)
(i)
(j)
(k)
(l)
(m)
(n)

Zadanie 1. Wykonaj krokową analizę działania algorytmu Kruskala dla powyższego grafu. (Pliki do wykorzystania: zadania_lab11.xlsx).

Zadanie 2. Znajdź minimalne drzewo rozpinające dla poniższego grafu. (Pliki do wykorzystania: zadania_lab11.xlsx).



ALGORYTM PRIMA

Algorytm Prima jest algorytmem zachłannym znajdującym minimalne drzewo rozpinające danego grafu spójnego z wagami.

Reguła zachłanna: dodaj krawędź o minimalnej wadze, która ma jeden wierzchołek w bieżącym drzewie a drugi, który nie należy do bieżącego drzewa.

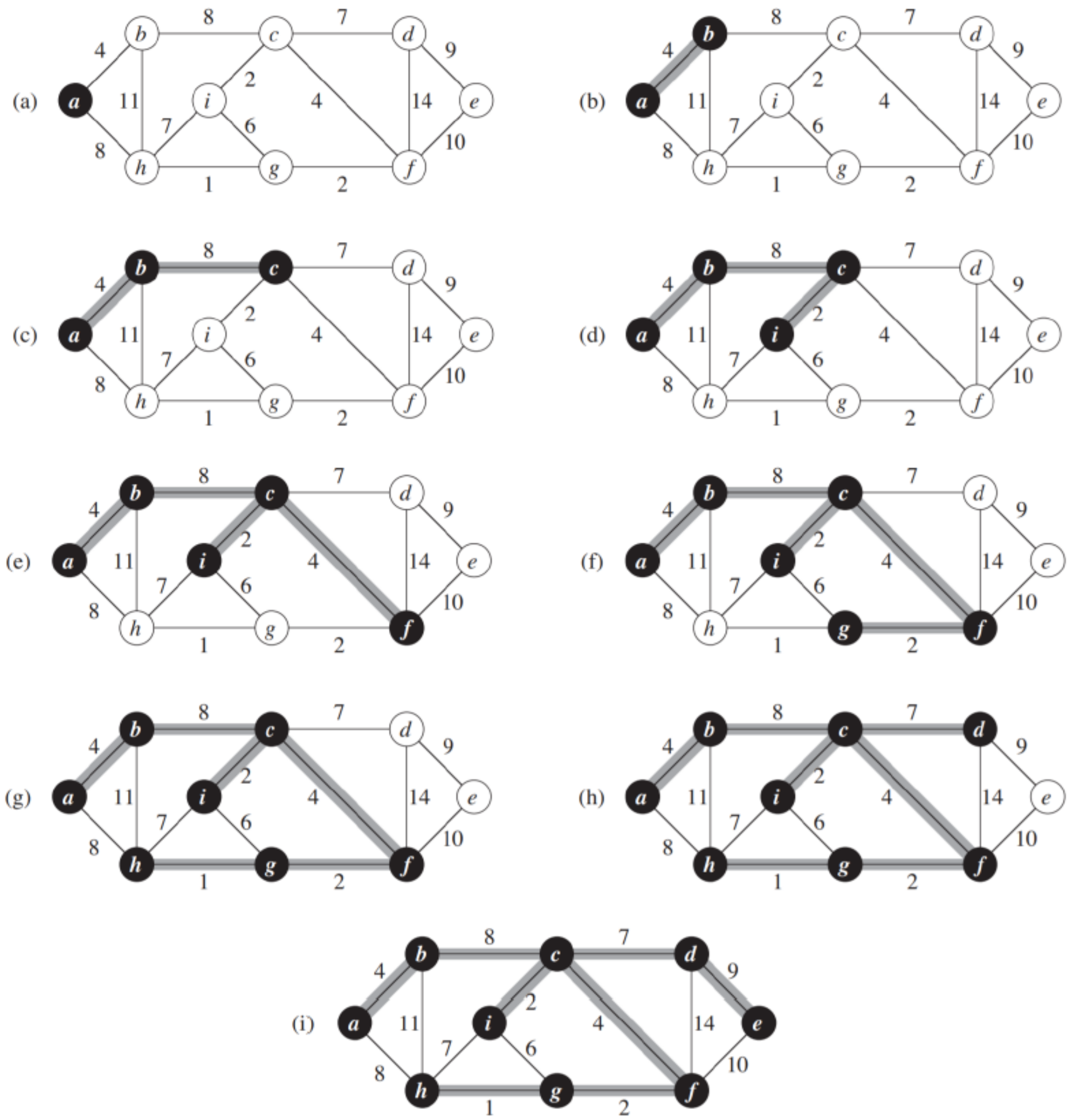
Każde rozwiązanie częściowe jest drzewem.

Wejście: G=(V,E,w) – graf spójny z wagami, s - wierzchołek startowy.

Wyjście: T – zbiór krawędzi minimalnego drzewa rozpinającego grafu G.

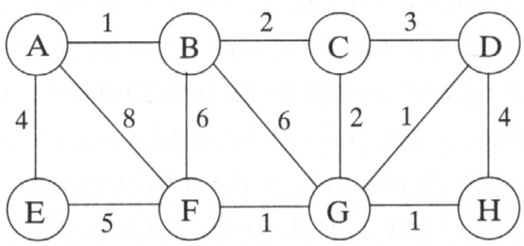
Algorytm:

1. T=∅
2. U={s}
3. while (U !=V)
4. znajdź krawędź (u,v)∈E o minimalnej wadze taką, że u∈U oraz v∈V−U;
5. T=T∪{(u,v)} ;
6. U=U∪{v} ;



Zadanie 3. Wykonaj krokową analizę działania algorytmu Prima dla powyższego grafu. (Pliki do wykorzystania: zadania_lab11.xlsx).

Zadanie 4. Chcemy znaleźć minimalne drzewo rozpinające dla poniższego grafu.



a) wykonaj algorytm Prima; za każdym razem, gdy pojawia się wybór wierzchołka, zawsze użyj tego, który jest pierwszy w kolejności alfabetycznej (startując od wierzchołka A).
b) Na tym samym grafie wykonaj algorytm Kruskala.

Zadanie 4. Zaproponuj schemat blokowych dla algorytmu Kruskala oraz Prima oraz ich implementacje.