

Zadanie 1

Przypuśćmy, że w grafie G wszystkie wagi krawędzi są różne. Pokaż, nie używając żadnego algorytmu, że G zawiera tylko jedno minimalne drzewo rozpinające.

Założmy, że dla G istnieją dwa minimalne drzewa rozpinające T_1 oraz T_2 . W jednym z tych drzew musi się znaleźć krawędź o minimalnej wadze spośród wszystkich wag. Bez straty ogólności, niech ta krawędź będzie w drzewie T_1 i oznaczmy ją przez e_1 . Wtedy graf $T_2 \cup \{e_1\}$ (nie drzewo, gdyż po dodaniu krawędzi e_1 do drzewa będzie n krawędzi) zawiera cykl i jedna z krawędzi tego cyklu, nazwijmy ją e_2 , nie należy do T_1 . Wiemy, że krawędzie e_1 i e_2 są różne i każda z nich należy dokładnie do jednego z drzew T_1 lub T_2 . Oznacza to, że $w(e_1) < w(e_2)$, gdyż e_1 jest najlżejszą wagą w drzewie T_1 , a e_2 najlżejszą w drzewie T_2 (gdyby tak nie było, to byśmy zaprzeczyli temu, że każda z tych krawędzi należy tylko do jednego minimalnego drzewa rozpinającego). Skoro tak, to $T = T_2 \cup \{e_1\} \setminus \{e_2\}$ jest drzewem rozpinającym. Łączna suma wag drzewa T jest mniejsza od łącznej sumy wag drzewa T_2 , więc dochodzimy do sprzeczności, gdyż założyliśmy, że T_2 jest minimalnym drzewem rozpinającym, co kończy dowód.

Zadanie 2

Niech T będzie MST grafu G . Pokaż, że dla dowolnego cyklu C grafu G drzewo T nie zawiera jakiegś najcięższej krawędzi z C .

Założmy nie wprost, że najcięższa krawędź e należy do T będącego minimalnym drzewem rozpinającym grafu G . Usunięcie tej krawędzi sprawiłoby, że T rozpadłoby się na dwa poddrzewa, których liśćmi byłyby wierzchołki tej krawędzi. Pozostała część cyklu C może połączyć ponownie te poddrzewa, więc musi istnieć jakaś krawędź $e_{new} \in C$ oparta na innych wierzchołkach. Oznacza to, że krawędź e_{new} łączy poddrzewa w drzewo T_2 o wagach mniejszych niż te, które znajdowały się w drzewie T , ponieważ $w(e_{new}) < w(e)$. Dochodzimy więc do sprzeczności, więc MST grafu G nie może zawierać najcięższej krawędzi cyklu C , co kończy dowód.

Zadanie 5

Założmy, że wszystkie krawędzie w grafie mają różne wagi. Udowodnij, że algorytm Boruvki rzeczywiście znajduje drzewo rozpinające, tzn. pokaż, że w żadnej iteracji nie powstaje cykl.

Algorytm Boruvki w normalnej wersji działa tak, że dla każdego wierzchołka grafu G wybieramy krawędź z najmniejszą wagą i dodajemy ją do zbioru E' . Jeżeli po tym kroku zostanie kilka spójnych składowych, zamieniamy je w *superwierzchołki* (czyli skleamy w jeden), a z pozostałymi krawędziami postępujemy tak, jakbyśmy zaczęli algorytm od nowa. Kroki te powtarzamy do momentu, aż w grafie pozostanie jedna spójna składowa.

Mamy udowodnić, że w każdej iteracji (jak również po zakończeniu) algorytmu Boruvki nie powstaje żaden cykl. Założmy nie wprost, że w którejś iteracji algorytmu pojawia się spójna składowa, w której jest cykl, nazwijmy ją A . Rozważmy dwie możliwości powstawania tego cyklu:

1. Spójna składowa A powstała przez połączenie dwóch superwierzchołków v_1, v_2 , czyli do zbioru E' zostały dodane jakieś krawędzie e_i, e_j . Zauważmy, że krawędź e_i była najlżejszą krawędzią incydentną do v_1 , a więc cykl C zawierający e_i będzie miał mniejszą wagę, niż cykl zawierający krawędź e_j . Ale skoro dodaliśmy do v_2 najlżejszą krawędź incydentną e_j , to cykl C zawierający e_j musi mieć mniejszą wagę, niż cykl zawierający e_i . Dochodzimy więc do sprzeczności, gdyż taka sytuacja nie jest możliwa.
2. Spójna składowa A powstała przez połączenie się kilku (trzech lub więcej) superwierzchołków. Podzielmy więc powstały cykl C w taki sposób: wierzchołki v_i dla $i = \{1, \dots, l\}$ są superwierzchołkami w tym cyklu C , a e_i dla $i = \{1, \dots, l\}$ są krawędziami dodanymi w kolejnych iteracjach algorytmu. Oznacza to, że krawędź $e_1 = \{v_1, v_2\}, e_2 = \{v_2, v_3\}, \dots, e_l = v_l, v_1$. Biorąc pod uwagę sposób, w jaki działa algorytm Boruvki, można stwierdzić, że aby powstał taki cykl C , musiałoby zachodzić

$$w(e_1) < w(e_2) < \dots < w(e_{l-1}) < w(e_l) < w(e_1)$$

więc dochodzimy do sprzeczności, gdyż $w(e_1) \not< w(e_1)$.

Udowodniliśmy więc, że algorytm Boruvki działa poprawnie i że w żadnej jego iteracji nie powstaje cykl.

Zadanie 6

Jak zmodyfikować algorytm Boruvki, by działał również w grafach, w których jakieś krawędzie mają takie same wagi?

Aby algorytm ten działał w grafach z krawędziami, które mają takie same wagi, musimy je w jakiś sposób rozróżnić. Poindeksujemy więc wszystkie krawędzie od i do $|E|$, a następnie dla takich samych wag krawędzi do wyboru, wybieramy za każdym razem tę z mniejszym indeksem. Dzięki temu krawędzie, których nie dodamy do E' przy pierwszym przejściu, zostaną dodane do MST w trakcie łączenia superwierzchołków (czyli spójnych składowych).

W podobny sposób można również posortować leksykograficznie wszystkie krawędzie: dla krawędzi $\{a, b\}$, $\{c, d\}$ o takich samych wagach, a więc $w(a, b) = w(c, d)$ (wierzchołki a, b, c, d nie muszą być różne). Wtedy możemy za każdym razem wybierać "mniejszą" krawędź na podstawie porządku leksykograficznego.

Zadanie 10

W pewnej grupie muzykujących osób Ania gra na skrzypcach, harfie, kontrabasie i wiolonczeli, Bartek gra na harfie i fortepianie, Cezary gra na fortepianie, Dąbrówka gra na harfie i Elwira gra na kontrabasie, skrzypcach, wiolonczeli i harfie. Chcieliby zagrać utwór na fortepian, skrzypce, wiolonczelę, kontrabas i harfę. Czy uda im się dobrać skład?

Przedstawmy skład i umiejętności wszystkich osób w tabelce:

	Ania	Bartek	Cezary	Dąbrówka	Elwira
skrzypce	X				X
harfa	X	X		X	X
kontrabas	X				X
wiolonczela	X				X
fortepian		X	X		

Aby dobrać skład, musimy zacząć od osób, które potrafią grać tylko na jednym instrumencie. Oznacza to, że Cezary musi grać na fortepianie, a Dąbrówka na harfie. Wtedy Bartek niezbyt się przyda reszcie, gdyż nie potrafi grać na żadnym z trzech brakujących instrumentów. Grupa zostaje więc z Anią i Elwirą potrafiącą grać na brakujących skrzypcach, kontrabasie i wiolonczeli, jednak dwie osoby raczej sobie nie poradzą z grą na trzech instrumentach jednocześnie. Oznacza to, że o ile Elwira i Ania nie są hiperuzdolnione, to nie uda się grupie zagrać.

Można również zauważyć, że

$$|\text{Anna, Elwira}| < |\text{wiolonczela, kontrabas, skrzypce}|$$

a więc nie jest spełniony warunek Halla, czyli się nie uda dobrać składu.