① Weźmy dowolny graf G i T_1, T_2 → jego MST ($T_1 \neq T_2$).

Weźmy krawędź e taką, że jest ona najlżejsza wśród krawędzi występujących albo w T_1, albo T_2. Niech $e \in T_1$.

Czyli $T_2 \cup \{e\}$ tworzy cykl c. Gdyby w T_1 były wszystkie wierzchołki cyklu c, to T_1 nie byłoby MST. Czyli istnieje jakaś krawędź d, której nie ma w T_1, a jest w T_2. Wiemy też, że $w(e) < w(d)$. Widać, że

$T_2 \cup \{e\} \setminus \{d\}$ ma mniejszą wagę od T_1 a też jest drzewem rozpinającym, czyli poprzez zastąpienie d przez e dostaniemy ST o mniejszej wadze.
Zatem sprzeczność → w G jest tylko jedno MST.

② Załóżmy, że najcięższa krawędź (e) cyklu C należy do T. Jeśli usuniemy ją z T, to otrzymamy dwa poddrzewa T_1, T_2. Ale $C \setminus \{e\}$ ma w sobie krawędź, która łączy ten rozbity cykl (i z tego łączy T_1 z T_2). Ale ta krawędź ma mniejszą wagę niż e, zatem znaleźliśmy drzewo rozpinające o mniejszej wadze, czyli $T \neq$ MST.

KIEDY WAGI RÓŻNE

~~Jeśli istnieją co najmniej dwie najcięższe krawędzie (np. e i d), to d nie należy do~~

⑤ Opis działania algorytmu:

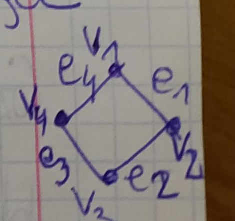
Dla każdego wierzchołka w G przeglądamy zbiór jego krawędzi i do zbioru E' dokładamy tę o najlżejszej wadze.
Tworzymy graf G' łącząc wierzchołki ze spójnych składowych w jeden (superwierzchołek).
Jeśli nie otrzymaliśmy jednej wspólnej składowej, to za G podstawiamy G' i wykonujemy algorytm jeszcze raz.

Dowód poprawności algorytmu (w każdym momencie po każdej iteracji algorytmu nie będzie cyklu)

Załóżmy, że gdzieś pojawi się składowa z cyklem S_C.

1) S_C pochodzi z dwóch superwierzchołków v_1, v_2 (i krawędzi e_1, e_2)
e_1 pochodzi jako najlejsza krawędź przy v_1 a e_2 jako najlżejsza przy v_2. Czyli waga cyklu z e_1 < waga cyklu z e_2, ale też cykl z e_2 będzie mieć mniejszą wagę niż ten z e_1. ($w(C(e_1)) < w(C(e_2)) \wedge w(C(e_1)) > w(C(e_2))$) ϟ.

2) S_C pochodzi z połączenia superwierzchołków trzech (lub więcej).
Mamy wierzchołki $v_1, \dots, v_k$ (kolejne superwierzchołki należące do C) i $e_1, \dots, e_k$ (kolejne krawędzie dodane do C).



Czyli, według algorytmu, musi zachodzić:

$$w(e_1) < w(e_2) < \dots < w(e_k) < w(e_1).$$

$w(e_1) < w(e_k) < w(e_1)$ daje sprzeczność.

Zatem w żadnym momencie działania algorytmu Borůvki nie powstanie cykl

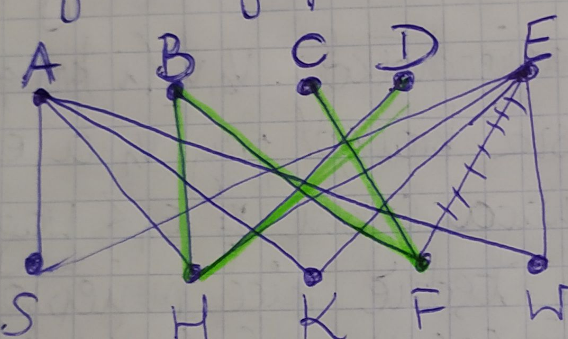
(6) Sposoby są co najmniej dwa.

Możemy najpierw przejść się po krawędziach i wybrać najmniejszą różnicę między wagami. Później podzielić tę różnicę przez **np.** 10000 i dla i-tej o tej samej wadze odjąć od tej wagi $i \cdot 10\,000$, w ten sposób, lekko zaburzając wagi, nie będą one takie same dla dwóch dowolnych krawędzi.

Można **też** jakoś ustawić krawędzie w tablicy i kiedy natrafimy na krawędzie o tej samej wadze, to porównywać ten indeks.

~~(11) Weźmy dowolny graf G~~

(10)



Tw. Graf dwudzielny ma skojarzenie doskonałe wtw. gdy spełniony w nim jest warunek Halla.

Czy warunek Halla jest spełniony? Otóż nie!

Weźmy $A' = \{B, C, D\} \subseteq \{A, B, C, D, E\}$

Wtedy $|N(A')| = |\{H, F\}| = 2$. No ale $|A'| = 3$.

Czyli, skoro $\neg(2 \geq 3)$, to warunek Halla niesp. $\Rightarrow$ grupa nie zagra tego utworu.