Algorytmy i Struktury Danych

Wykład 6

Zastosowania algorytmu DFS

1 Sortowanie topologiczne DAGu

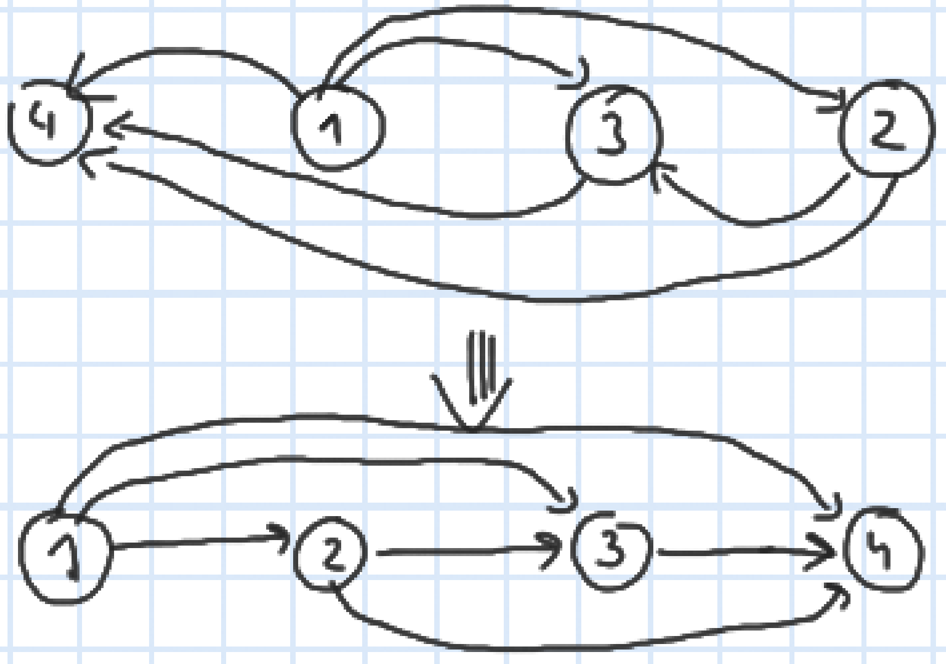
DAG – directed acyclic graph

Sortowanie topologiczne – ułożenie wierzchołków grafu w takiej kolejności, że krawędzie wskazują tylko "z lewej na prawą"

Przykłady zastosowania

– wyznaczanie kolejności realizacji zadań

– linearyzacja częściowego porządku

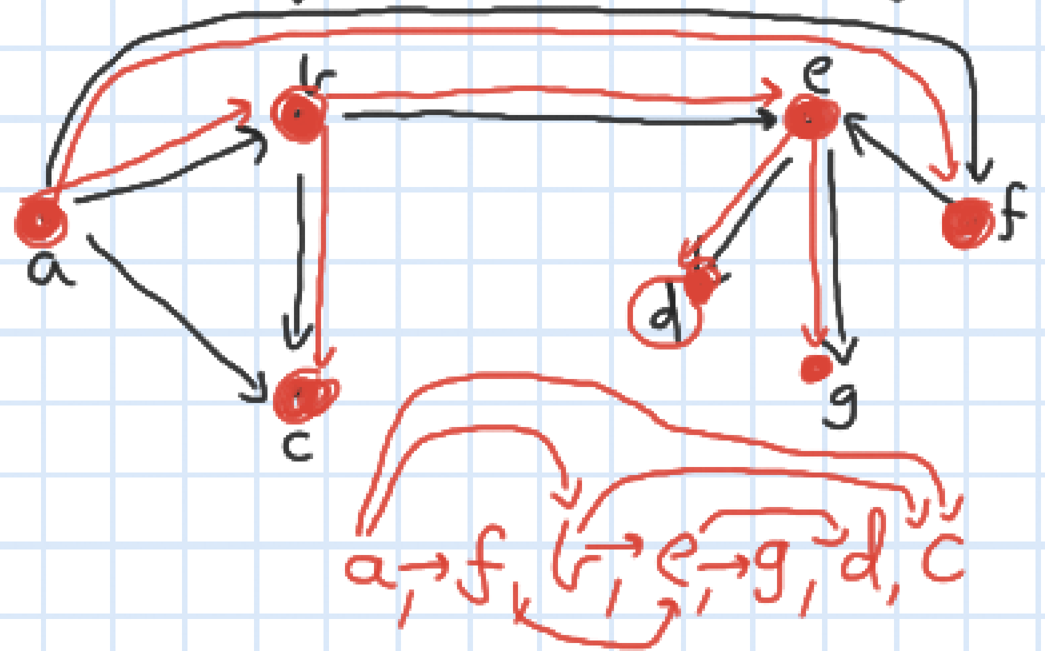


Algorytm

– uruchamiamy DFS

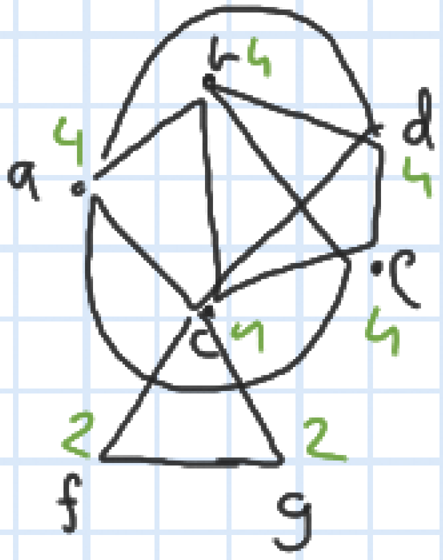
– po "przetworzeniu" danego wierzchołka dopisujemy go na początek tworzonej listy

– lista daje posortowaną kolejność



② Cykl Eulera

Cykl Eulera to cykl przebiegający przez każdą krawędź w grafie



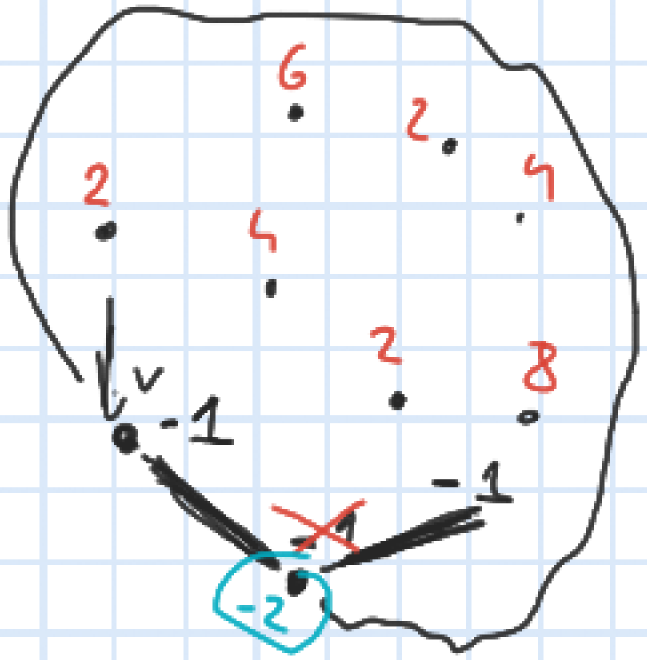
tw Graf nieskierowany, spójny posiada cykl Eulera wtw gdy każdy wierzchołek ma stopień parzysty

dowód

Warunek konieczny – oczywiste

Warunek wystarczający:

- zaczynając od dowolnie wybranego wierzchołka v, wędrujemy dowolnie wybranymi krawędziami (bez powtarzania)



- w pewnym momencie musimy wrócić do v

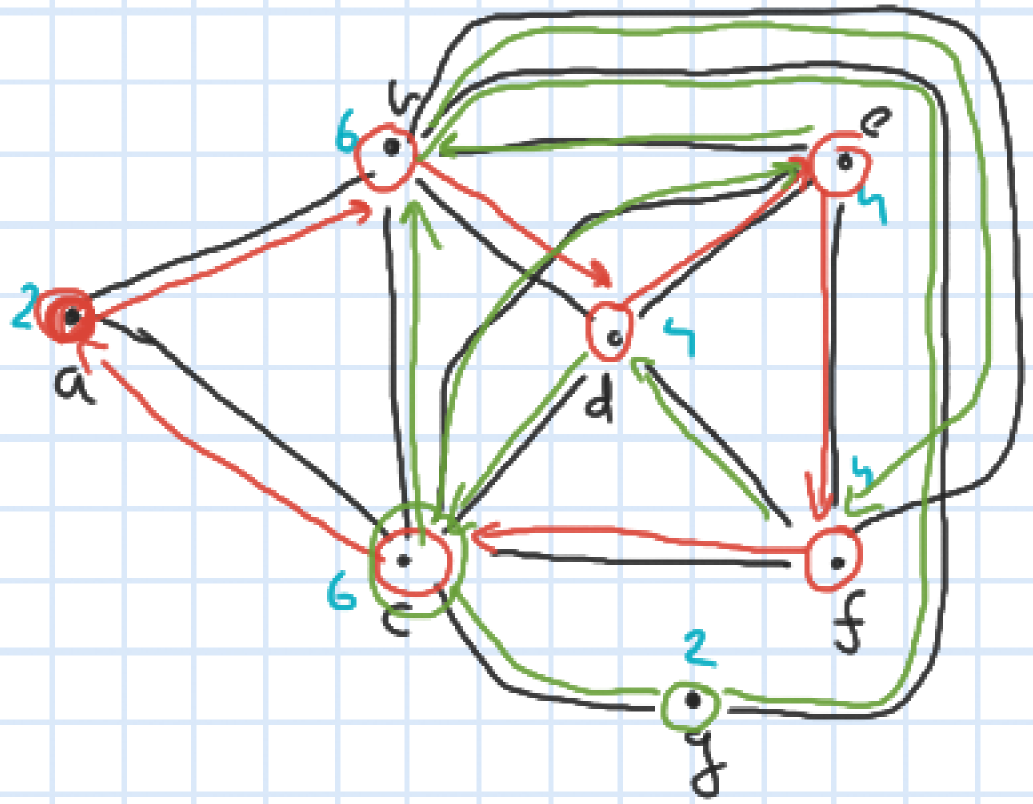
jeśli odwiedziliśmy wszystkie krawędzie, to mamy cykl Eulera

jeśli , to możemy "dokleić" fragment cyklu startując od nowego wierzchołka, z którego wychodzą nieodwiedzone krawędzie

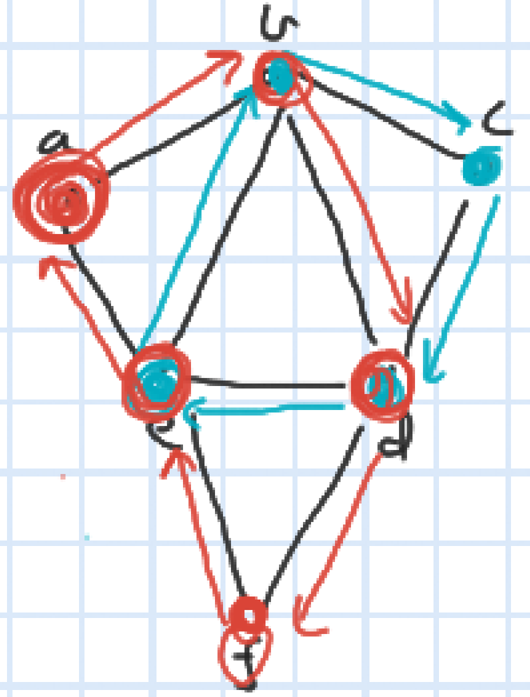
Algorytm

- wykonujemy DFS, "usuwając" na bieżąco odwiedzane krawędzie i nie zabraniając wielokrotnego wejścia do tego samego wierzchołka

- po przetworzeniu wierzchołka dodajemy go na początek tworzonego cyklu



a b d e f c b f d c e b g c a



a b d f e b c d e a

Cykl Hamiltona — cykl przebiegający dokładnie raz przez każdy wierzchołek grafu

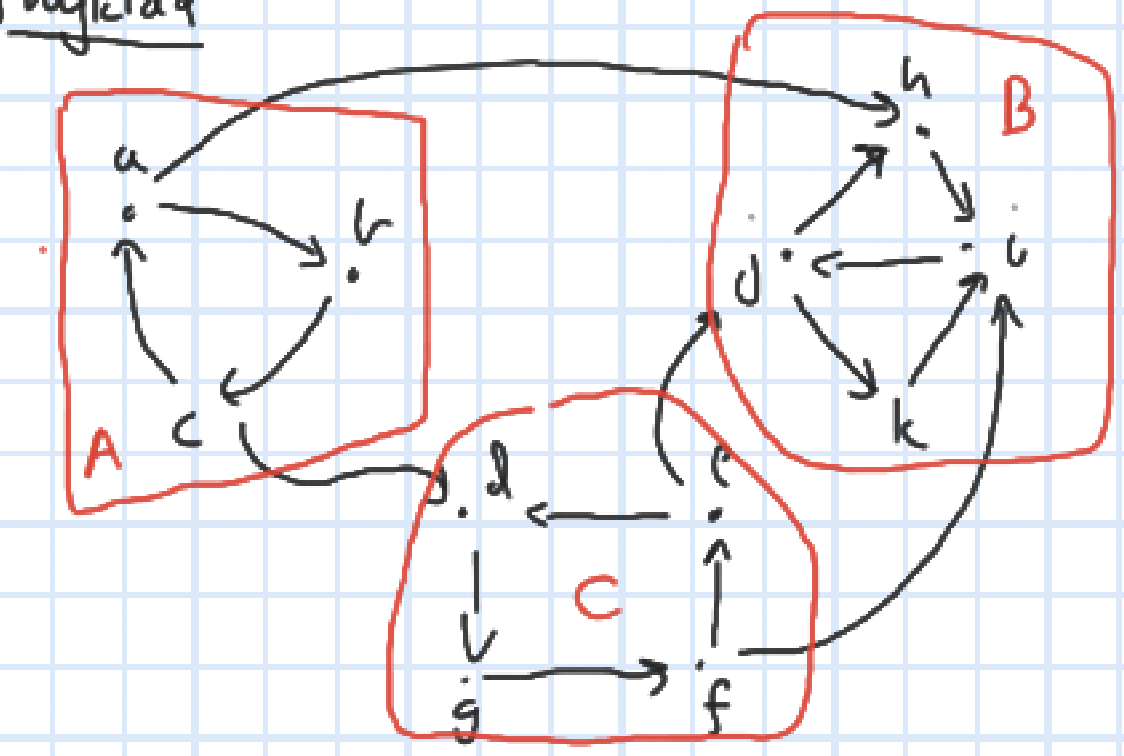
↓

stwierdzenie czy cykl Hamiltona istnieje jest problemem NP-zupełnym

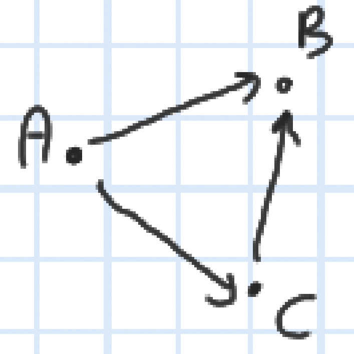
③ Silnie spójne składowe w grafie skierowanym

def Niech $G=(V,E)$ będzie grafem skierowanym. Wierzchołki u i v należą do tej samej silnie spójnej składowej jeśli są wzajemnie osiągalne ścieżkami zgodnie z kierunkiem krawędzi

Przykład



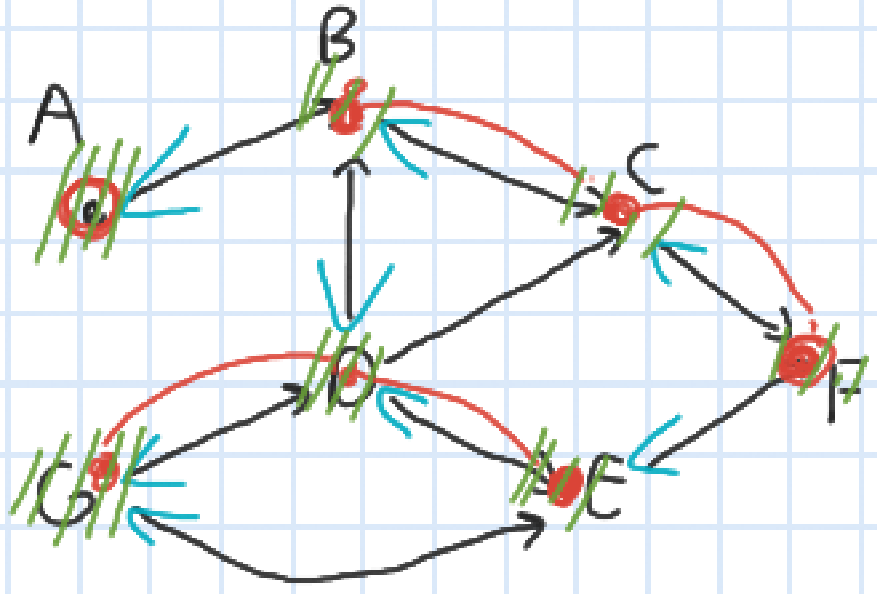
silnie spójne składowe tworzą DAG

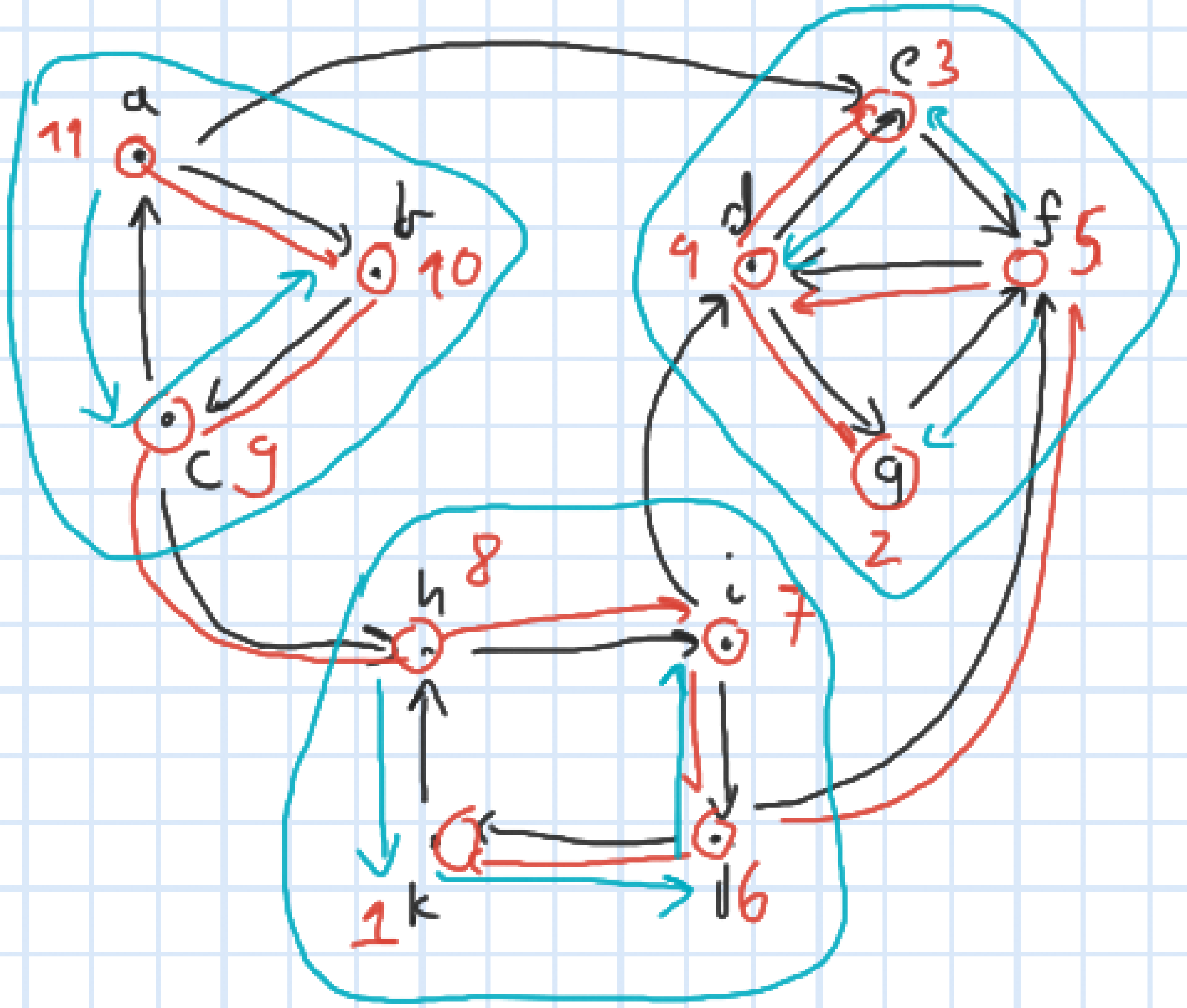


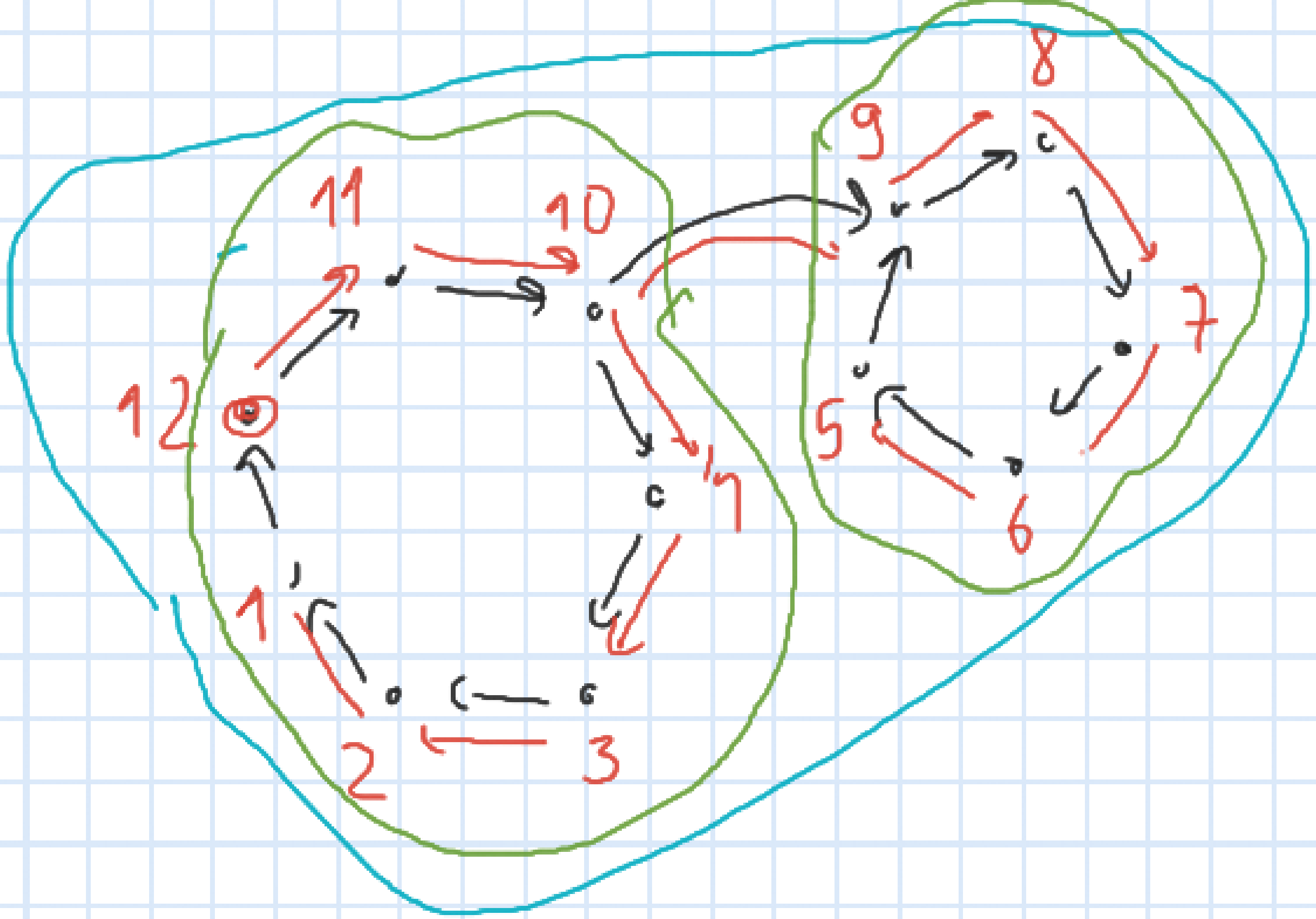
Algorytm

1. Wykonaj DFS zapisując czasy przetworzenia wierzchołków
2. Odwróć kierunek krawędzi
3. Wykonaj DFS ponownie, w kolejności malejących czasów przetworzenia z pierwszego DFS

Intuicja




a 11
b 10
c 9
d 4
e 3
f 5
g 2
h 8
i 7
k 1
6


12
11
10
4
1
2
3
9
8
7
5
6

④ Mosty w grafach nieskierowanych

def $G=(V,E)$ - graf nieskierowany. Krawędź $e \in E$ nazywamy mostem jeśli jej usunięcie spowoduje, że graf stanie się niespójny

tw Krawędź e jest mostem wtw gdy nie leży na żadnym cyklu prostym w grafie

dowód

$e=\{u,v\}$ jest mostem $\Rightarrow$ e nie leży na cyklu bo inaczej po jej usunięciu wciąż byłaby ścieżka z u do v

$e=\{u,v\}$ nie leży na żadnym cyklu $\Rightarrow$ jest mostem bo po jej usunięciu nie ma drogi z u do v

Algorytm

1. Wykonaj DFS, zapisując dla każdego wierzchołka v czas odwiedzenia $d(v)$

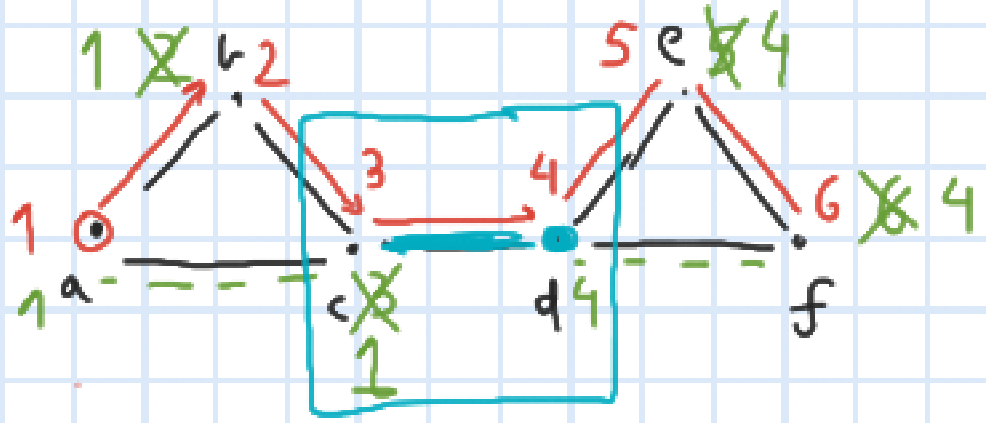
2. Dla każdego wierzchołka v oblicz:

$$low(v) = \min\Big(d(v),\ \min_{\substack{u \text{ - jest wierzchołkiem} \\ \text{osiągalnym krawędzią} \\ \text{wsteczną z } v}} d(u),\ \min_{\substack{w \text{ jest dzieckiem } v \\ \text{w drzewie DFS}}} low(w) \Big)$$

krawędź do odwiedzonego, nie przetworzonego wierzchołka (poza krawędzią do rodzica)

3. Mosty to krawędzie $\{v, p(v)\}$ gdzie $d(v) = low(v)$

rodzic v



d(v)

low(v)