

## 1) Grafy eulerowskie

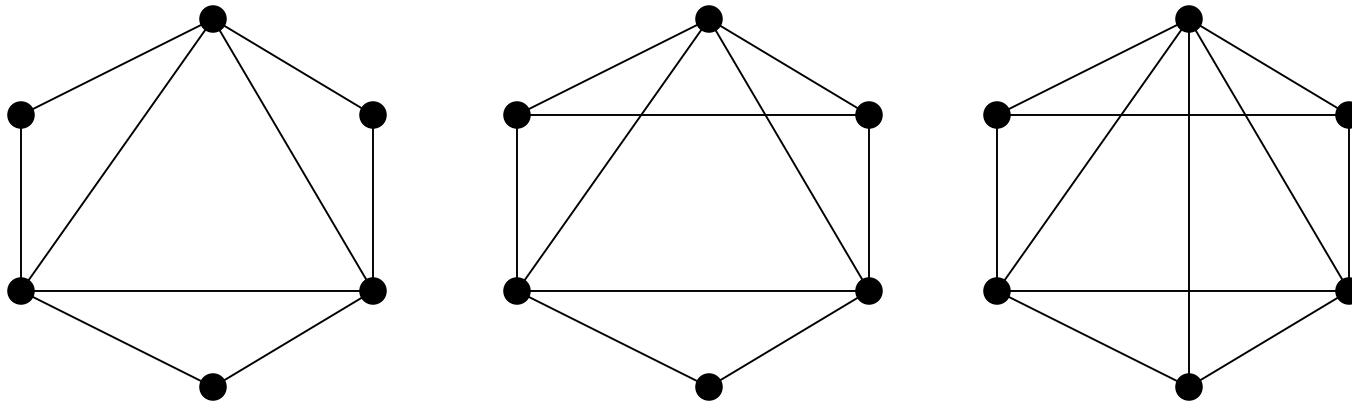
- własności
- algorytmy

## 2) Problem chińskiego listonosza

# Grafy eulerowskie

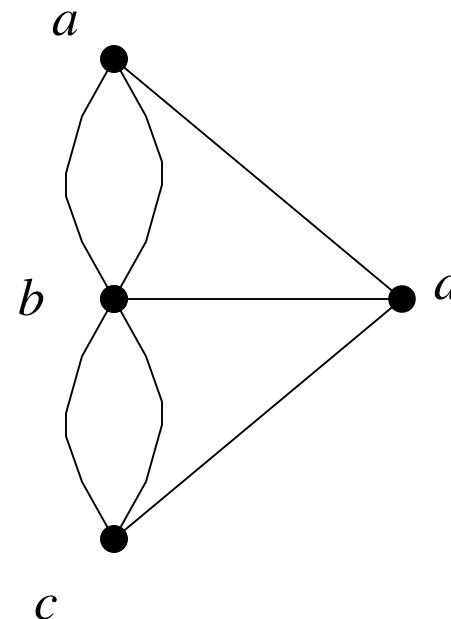
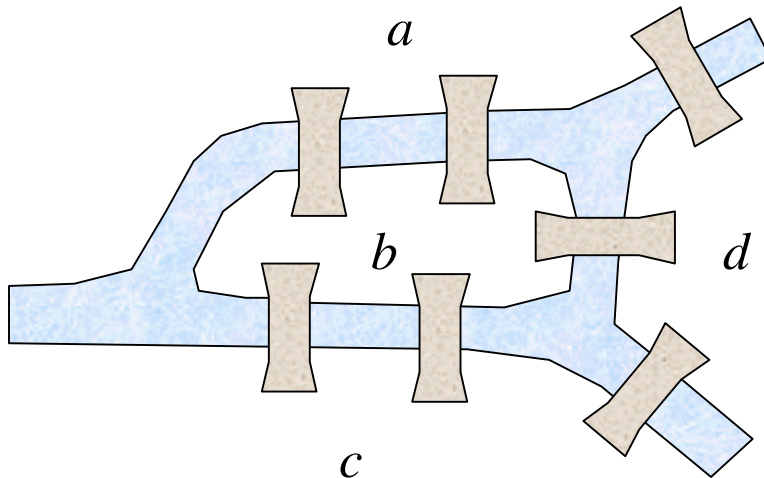
**Def.** Graf (multigraf, niekoniecznie spójny) jest *grafem eulerowskim*, jeśli zawiera cykl zawierający wszystkie krawędzie. Graf jest *półeulerowski*, gdy posiada łańcuch o powyższej własności.

**Przykład** Poniżej podano reprezentacje trzech grafów, z których pierwszy jest eulerowski, drugi – półeulerowski, natomiast ostatni nie jest ani eulerowski ani półeulerowski.



# Grafy eulerowskie

Nazwa „eulerowski” pochodzi stąd, iż Euler w 1736 r. rozwiązał „problem mostów królewieckich”. Pytano, czy można przejść dokładnie raz przez każdy z siedmiu mostów (rys. po lewej) tak, aby powrócić do punktu wyjścia. Zauważmy, że problem jest równoważny stwierdzeniu, czy graf pokazany na rys. po prawej stronie jest eulerowski.



# Grafy eulerowskie

Aby podać warunek konieczny i dostateczny na to, aby dany graf był eulerowski będzie potrzebny poniższy lemat:

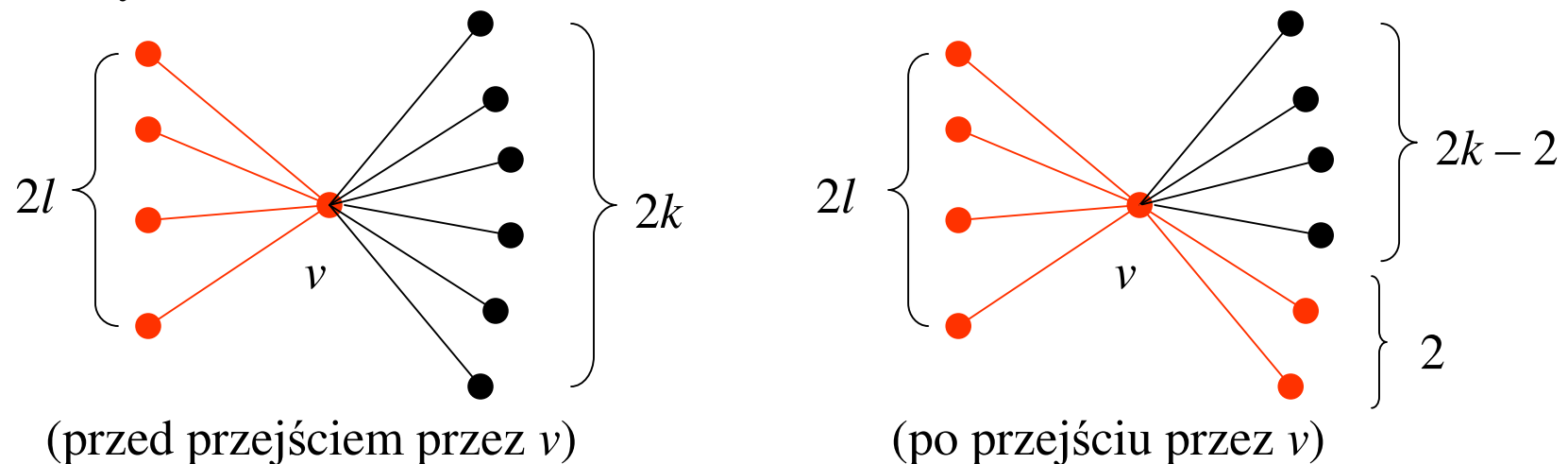
**Lemat** *Jeśli każdy wierzchołek multigrafu  $G$  ma stopień równy co najmniej 2, to  $G$  zawiera cykl.*

**Dowód:** Jeśli  $G$  zawiera pętle lub krawędzie wielokrotne, to lemat jest spełniony, więc zakładamy dalej, że  $G$  jest grafem prostym. Następnie konstruujemy ścieżkę: wybieramy dowolny wierzchołek  $v_1 \in V(G)$ , po czym dodajemy kolejne według zasady: jeśli  $v_i$  jest ostatnio dodanym wierzchołkiem, to jeśli  $v_i$  jest sąsiedni z  $v_j$  poprzednio dodanym do ścieżki, to otrzymujemy cykl zawierający  $v_j, \dots, v_i$  co kończy dowód. W przeciwnym wypadku wybieramy dowolnego sąsiada wierzchołka  $v_i$  (istnieje, ponieważ  $\deg(v_i) > 1$ ) i dodajemy go do ścieżki. Graf  $G$  jest skończony więc w pewnym kroku otrzymamy wierzchołek  $v_k$ , który jest sąsiedni z pewnym  $v_l$ , gdzie  $l < k$ .

# Tw. Eulera

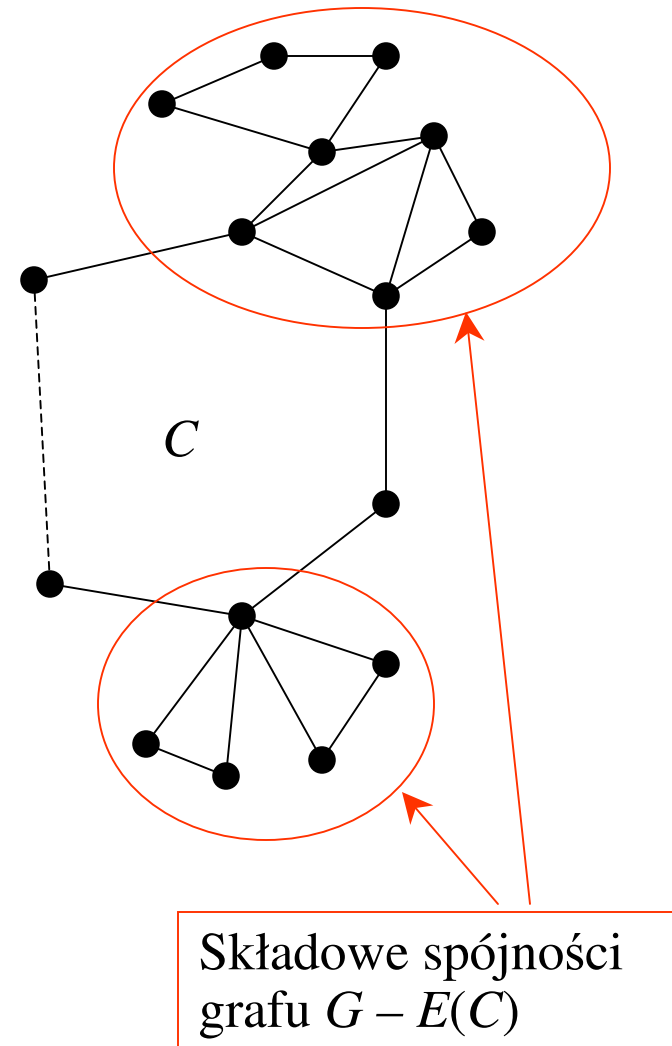
**Tw. (Euler, 1736)** *Multigraf spójny jest eulerowski wtedy i tylko wtedy, gdy stopień każdego wierzchołka jest liczbą parzystą.*

**Dowód:** ( $\Rightarrow$ ) Jeśli  $C$  jest cyklem Eulera w  $G$ , to trawersujemy i usuwamy krawędzie w  $E(G)$  zgodnie z kolejnością zadaną przez  $C$ . Gdy przechodzimy przez dowolny wierzchołek  $v$ , to usuwamy dwie incydentne z nim krawędzie, więc stopień tego wierzchołka w tak zredukowanym grafie pozostaje parzysty. Na rys. poniżej kolorem czerwonym oznaczono krawędzie, które zostały usunięte z  $G$ .



# Tw. Eulera

**Dowód tw. Eulera** ( $\Leftrightarrow$ )  $G$  jest spójny, więc dla każdego  $v$  zachodzi  $\deg(v) > 1$  więc z poprzedniego lematu wiadomo, że  $G$  zawiera cykl, który oznaczmy przez  $C$ . Twierdzenie dowodzimy przez indukcję względem  $m$ . Jeśli  $C = G$ , to dowód jest zakończony. W przeciwnym wypadku każda składowa spójności grafu  $G - E(C)$  spełnia założenie twierdzenia, więc z założenia indukcyjnego jest eulerowska. Znajdujemy cykl Eulera w  $G$  następująco: przechodzimy przez kolejne wierzchołki w  $C$  i jeśli bieżący wierzchołek należy do pewnej składowej spójności, to trawersujemy cykl Eulera w tej składowej, powracając do tego samego wierzchołka w  $C$ .



# Algorytm Fleury'ego

**Wniosek** *Multigraf spójny jest eulerowski wtedy i tylko wtedy, gdy zbiór jego krawędzi można podzielić na rozłączne cykle.*

**Wniosek** *Multigraf spójny jest półeulerowski, gdy posiada co najwyżej dwa wierzchołki nieparzystego stopnia. Jeden z nich jest początkiem, a drugi końcem łańcucha Eulera.*

**Algorytm Fleury'ego** (znajdujący cykl Eulera)

- rozpocznij wędrówkę w dowolnym wierzchołku
- usuwaj strawersowane krawędzie, przechodząc po moście jedynie w ostateczności (sytuacja, w której są do wyboru co najmniej dwa mosty oznacza, że graf nie jest eulerowski).

# Inny algorytm...

```

procedure Euler(  $G$  )
begin
   $stos := \emptyset$ ;  $ce := \emptyset$ ;
   $stos.push(\text{dowolny wierzchołek grafu})$ ;
  while  $stos \neq \emptyset$  do begin
     $v := stos.top$ ;
    if  $deg(v) \neq 0$  then begin
       $u$  – sąsiad  $v$  o najniższym indeksie;
       $G := (V(G), E(G) \setminus \{u, v\})$ ;
       $stos.push(u)$ ;
    end else begin (*  $deg(v) = 0$  *)
       $stos.pop(v)$ ;
       $ce.push(v)$ ;
    end
  end
end

```

## Zmienne:

$stos$  – stos pomocniczy;  
 $ce$  – stos przechowujący  
 wierzchołki w kolejności  
 w jakiej tworzą cykl Eulera

## Uwagi:

- operacja *top* zwraca wierzchołek na szczycie stosu, ale zawartość stosu się nie zmienia
- wynik działania procedury to cykl Eulera zapamiętany w zmiennej *ce*
- *deg* zwraca stopień wierzchołka w zredukowanym grafie

# Problem chińskiego listonosza

Dana jest sieć ulic oraz poczta. Aby listonosz dostarczył korespondencję musi przejść wzdłuż każdej ulicy co najmniej raz i powrócić do punktu wyjścia. Formułując problem w języku grafów, pytamy o najkrótszą zamkniętą marszrutę w grafie  $G$  utworzonym na podstawie sieci ulic, w którym wagi krawędzi odpowiadają długościom ulic.

Znany jest efektywny algorytm rozwiązujący ten problem. Rozważymy trzy przypadki:

*Przypadek 1:* graf  $G$  jest eulerowski. Wówczas każdy cykl Eulera jest optymalnym rozwiązaniem, które można znaleźć korzystając np. z algorytmu Fleury'ego.

# CPP: przypadek 2

Graf  $G$  jest półeulerowski. Znajdujemy ścieżkę Eulera łączącą dwa wierzchołki nieparzystego stopnia  $u$  i  $v$ . Następnie szukamy najkrótszej drogi z  $u$  do  $v$ . Łącząc obie drogi otrzymujemy rozwiązanie.

## Przykład

### 1) Droga Eulera:

$$b \rightarrow a \rightarrow f \rightarrow b \rightarrow c \rightarrow$$

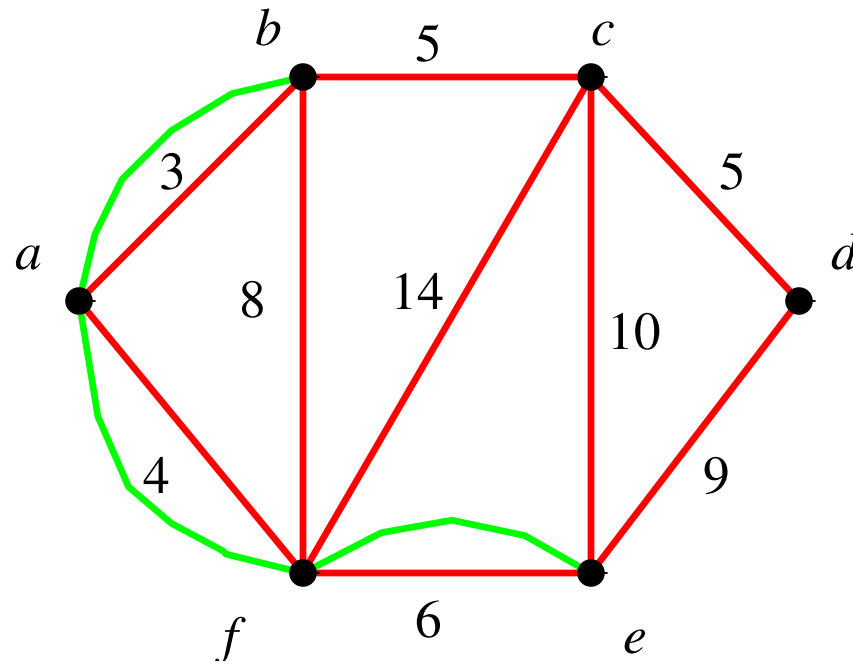
$$d \rightarrow e \rightarrow c \rightarrow f \rightarrow e$$

### 2) Najkrótsza droga z $e$ do $b$

$$e \rightarrow f \rightarrow a \rightarrow b$$

### 3) „Trasa listonosza”:

$$b \rightarrow a \rightarrow f \rightarrow b \rightarrow c \rightarrow$$

$$d \rightarrow e \rightarrow c \rightarrow f \rightarrow e \rightarrow f \rightarrow a \rightarrow b$$


# CPP: przypadek ogólny

1. Zidentyfikuj wierzchołki nieparzystego stopnia w grafie  $G$ . Niech  $W$  będzie zbiorem takich wierzchołków.
2. Skonstruuj obciążony graf pełny  $G'$  o zbiorze wierzchołków  $W$ , w którym waga krawędzi  $\{u, v\}$  jest równa długości najkrótszej ścieżki łączącej  $u$  z  $v$  w wyjściowym grafie  $G$ .

