

3-10 Traversability

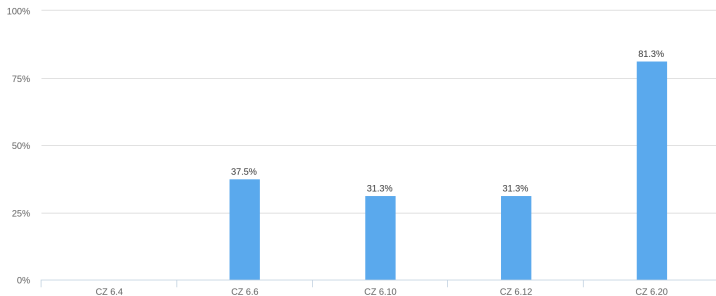
(Part I: Eulerian Graphs)

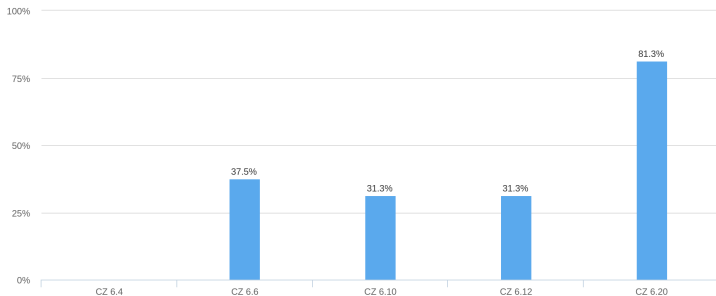
Hengfeng Wei

hfwei@nju.edu.cn

December 03, 2018







CZ 6.20 (Next Class)

这次习题相对简单【因为是最难的所以随机选了一个】，希望老师可以多回顾一下课本内容，比如哈密尔顿图的各种充分条件和证明，对哈密尔顿图和欧拉图的应用做一些拓展

欧拉图和哈密尔顿图的联系 就是在建模时如何确定图的节点和边

g-cage 对于不同大小而言都是唯一的吗？（书上只给到 $n=8$ ） Kirkman triple system 和 factorization 的等价推导 Theorem 8.10的证明

6.3, 不想看书，看自闭了，希望老师可以直接讲一下

如何打印欧拉回路/欧拉迹/哈密尔顿回路

定理6.5

无

暂无

无

陶老师上课讲的中国邮递员问题没明白

暂无，我就来抽个奖

none

定理6.12的证明

喵(^ _ ^)

希望能讲一下fluery算法

可以总结一下证明的方法，其实每次都可以这样，不一定要课上讲，可以整理之后做成讲义课后发，比如怎么证明有欧拉回路等等

这次习题相对简单【因为是最难的所以随机选了一个】，希望老师可以多回顾一下课本内容，比如哈密尔顿图的各种充分条件和证明，对哈密尔顿图和欧拉图的应用做一些拓展

欧拉图和哈密尔顿图的关系 就是在建模时如何确定图的节点和边

g-cage 对于不同大小而言都是唯一的吗？（书上只给到 $n=8$ ） Kirkman triple system 和 factorization 的等价推导 Theorem 8.10的证明

6.3, 不想看书，看自闭了，希望老师可以直接讲一下

如何打印欧拉回路/欧拉迹/哈密尔顿回路

定理6.5

无

暂无

无

陶老师上课讲的中国邮递员问题没明白

暂无，我就来抽个奖

none

定理6.12的证明

喵(^ _ ^)

希望能讲一下fleury算法

可以总结一下证明的方法，其实每次都可以这样，不一定要课上讲，可以整理之后做成讲义课后发，比如怎么证明有欧拉回路等等

FLEURY

这次习题相对简单【因为是最难的所以随机选了一个】，希望老师可以多回顾一下课本内容，比如哈密尔顿图的各种充分条件和证明，对哈密尔顿图和欧拉图的应用做一些拓展

欧拉图和哈密尔顿图的关系 就是在建模时如何确定图的节点和边

g-cage 对于不同大小而言都是唯一的吗？（书上只给到 $n=8$ ） Kirkman triple system 和 factorization 的等价推导 Theorem 8.10的证明

6.3, 不想看书，看自闭了，希望老师可以直接讲一下

如何打印欧拉回路/欧拉迹/哈密尔顿回路

定理6.5

无

暂无

无

陶老师上课讲的中国邮递员问题没明白

暂无，我就来抽个奖

none

定理6.12的证明

喵(^ _ ^)

希望能讲一下fleury算法☺

可以总结一下证明的方法，其实每次都可以这样，不一定要课上讲，可以整理之后做成讲义课后发，比如怎么证明有欧拉回路等等

FLEURY (HIERHOLZER)

这次习题相对简单【因为是最难的所以随机选了一个】，希望老师可以多回顾一下课本内容，比如哈密尔顿图的各种充分条件和证明，对哈密尔顿图和欧拉图的应用做一些拓展

欧拉图和哈密尔顿图的关系 就是在建模时如何确定图的节点和边

g-cage 对于不同大小而言都是唯一的吗？（书上只给到 $n=8$ ） Kirkman triple system 和 factorization 的等价推导 Theorem 8.10的证明

6.3, 不想看书，看自闭了，希望老师可以直接讲一下

如何打印欧拉回路欧拉迹/哈密尔顿回路

定理6.5

无

暂无

无

陶老师上课讲的中国邮递员问题没明白

暂无，我就来抽个奖

none

定理6.12的证明

喵(^ . ^)

希望能讲一下fleury算法

可以总结一下证明的方法，其实每次都可以这样，不一定要课上讲，可以整理之后做成讲义课后发，比如怎么证明有欧拉回路等等

FLEURY (HIERHOLZER)

Chinese Postman Problem (Next Class)

这次习题相对简单【因为是最难的所以随机选了一个】，希望老师可以多回顾一下课本内容，比如哈密尔顿图的各种充分条件和证明，对哈密尔顿图和欧拉图的应用做一些拓展

欧拉图和哈密尔顿图的关系 就是在建模时如何确定图的节点和边

g-cage 对于不同大小而言都是唯一的吗？（书上只给到 $n=8$ ） Kirkman triple system 和 factorization 的等价推导 Theorem 8.10的证明

6.3, 不想看书，看自闭了，希望老师可以直接讲一下

如何打印欧拉回路/欧拉迹/哈密尔顿回路

定理6.5

无

暂无

无

向老师上课讲的中国邮递员问题没明白

暂无，我就来抽个奖

none

定理6.12的证明

喵(=^_^=)

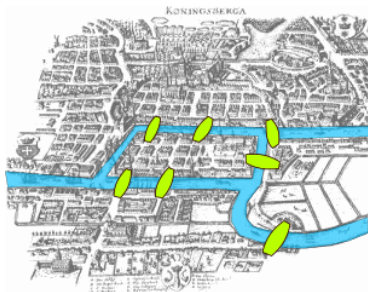
希望能讲一下fleury算法☺

可以总结一下证明的方法，其实每次都可以这样，不一定要课上讲，可以整理之后做成讲义课后发，比如怎么证明有欧拉回路等等

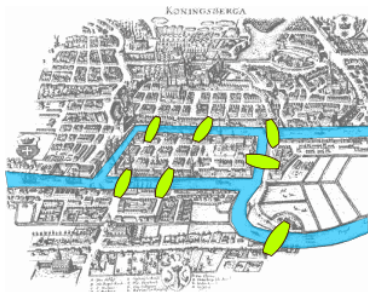
FLEURY (HIERHOLZER)

Chinese Postman Problem (Next Class)

6.3 Exploration & 6.4 Excursion (Not Required)



Leonhard Euler (1707 – 1783)

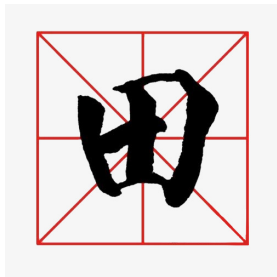


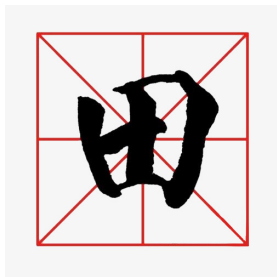
Leonhard Euler (1707 – 1783)

Graph Theory

Topology







Theorem (Leonhard Euler 1735)

*A connected graph G is **Eulerian** if and only if the degree of each vertex of G is even.*

Theorem (Leonhard Euler 1735)

*A connected graph G is **Eulerian** if and only if the degree of each vertex of G is even.*

“ \Leftarrow ” (Carl Hierholzer 1873);

Theorem (Leonhard Euler 1735)

*A connected graph G is **Eulerian** if and only if the degree of each vertex of G is even.*

“ \Leftarrow ” (Carl Hierholzer 1873); (Proofs on PPT from Tao & on CZ).

Theorem (Leonhard Euler 1735)

A connected graph G is *Eulerian* if and only if the degree of each vertex of G is even.

“ \Leftarrow ” (Carl Hierholzer 1873); (Proofs on PPT from Tao & on CZ).

By strong mathematical induction on the number m of edges of G .

Theorem (Leonhard Euler 1735)

A connected graph G is *Eulerian* if and only if the degree of each vertex of G is even.

“ \Leftarrow ” (Carl Hierholzer 1873); (Proofs on PPT from Tao & on CZ).

By strong mathematical induction on the number m of edges of G .

Inductive Step : $m = k + 1$

Theorem (Leonhard Euler 1735)

A connected graph G is *Eulerian* if and only if the degree of each vertex of G is even.

“ \Leftarrow ” (Carl Hierholzer 1873); (Proofs on PPT from Tao & on CZ).

By strong mathematical induction on the number m of edges of G .

Inductive Step : $m = k + 1$

Let $C = u \sim v$ be any trail in G of the maximum length.

Theorem (Leonhard Euler 1735)

A connected graph G is *Eulerian* if and only if the degree of each vertex of G is even.

“ \Leftarrow ” (Carl Hierholzer 1873); (Proofs on PPT from Tao & on CZ).

By strong mathematical induction on the number m of edges of G .

Inductive Step : $m = k + 1$

Let $C = u \sim v$ be any trail in G of the maximum length.

$$v = u$$

Theorem (Leonhard Euler 1735)

A connected graph G is *Eulerian* if and only if the degree of each vertex of G is even.

“ \Leftarrow ” (Carl Hierholzer 1873); (Proofs on PPT from Tao & on CZ).

By strong mathematical induction on the number m of edges of G .

Inductive Step : $m = k + 1$

Let $C = u \sim v$ be any trail in G of the maximum length.

$$v = u \implies C = u \sim u$$

Theorem (Leonhard Euler 1735)

A connected graph G is *Eulerian* if and only if the degree of each vertex of G is even.

“ \Leftarrow ” (Carl Hierholzer 1873); (Proofs on PPT from Tao & on CZ).

By strong mathematical induction on the number m of edges of G .

Inductive Step : $m = k + 1$

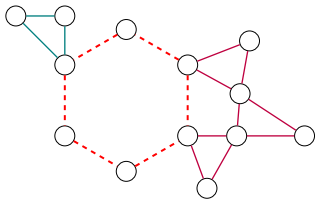
Let $C = u \sim v$ be any trail in G of the maximum length.

$$v = u \implies C = u \sim u$$

$$H = G - E(C) = \bigcup H_i$$

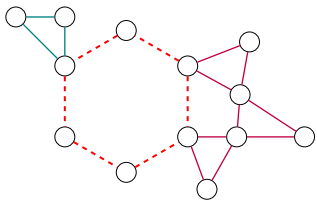


$$H = G - E(C) = \bigcup H_i$$



$$H = G - E(C) = \bigcup H_i$$

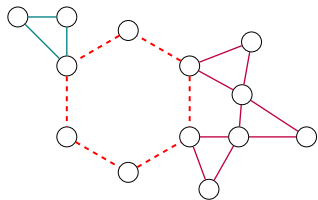
- (I) $\forall v \in H : \deg(v)$ is even
- (II) $\forall i : |E(H_i)| < m$



$$H = G - E(C) = \bigcup H_i$$

(I) $\forall v \in H : \deg(v)$ is even

(II) $\forall i : |E(H_i)| < m$

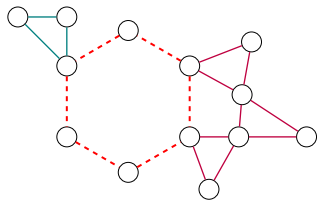


By I.H., each H_i has an Eulerian circuit C_i .

$$H = G - E(C) = \bigcup H_i$$

(I) $\forall v \in H : \deg(v)$ is even

(II) $\forall i : |E(H_i)| < m$



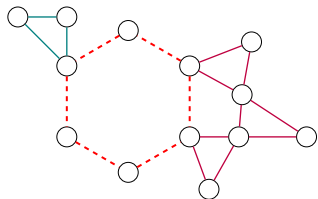
By I.H., each H_i has an Eulerian circuit C_i .

$$\forall i : V(H_i) \cap V(C) \neq \emptyset$$

$$H = G - E(C) = \bigcup H_i$$

(I) $\forall v \in H : \deg(v)$ is even

(II) $\forall i : |E(H_i)| < m$



By I.H., each H_i has an Eulerian circuit C_i .

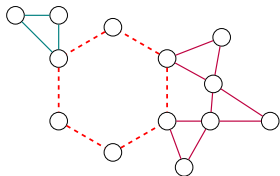
$$\forall i : V(H_i) \cap V(C) \neq \emptyset$$

Glue together each C_i with C to get an Eulerian circuit of G .

```

1: procedure HIERHOLZER( $G$ )
2:    $u \in V(G)$ 
3:    $C \leftarrow$  any circuit  $u \sim u$  in  $G$ 
4:   while  $\exists v \in C : \deg(v) > 0$  do
5:      $H \leftarrow G - E(C)$ 
6:      $v \leftarrow$  any vertex in  $V(C)$  such that  $\deg(v) > 0$ 
7:      $C' \leftarrow$  any circuit  $v \sim v$  in  $H$ 
8:      $C \leftarrow C \otimes C'$  ▷ Glue  $C' = v \sim v$  with  $C$  via  $v$ 
9:   return  $C$ 

```

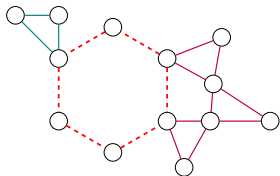


Q : Time Complexity?

```

1: procedure HIERHOLZER( $G$ )
2:    $u \in V(G)$ 
3:    $C \leftarrow$  any circuit  $u \sim u$  in  $G$ 
4:   while  $\exists v \in C : \deg(v) > 0$  do
5:      $H \leftarrow G - E(C)$ 
6:      $v \leftarrow$  any vertex in  $V(C)$  such that  $\deg(v) > 0$ 
7:      $C' \leftarrow$  any circuit  $v \sim v$  in  $H$ 
8:      $C \leftarrow C \otimes C'$  ▷ Glue  $C' = v \sim v$  with  $C$  via  $v$ 
9:   return  $C$ 

```



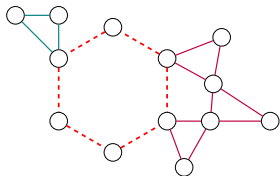
Q : Time Complexity?

Q : Data Structures?

```

1: procedure HIERHOLZER( $G$ )
2:    $u \in V(G)$ 
3:    $C \leftarrow$  any circuit  $u \sim u$  in  $G$ 
4:   while  $\exists v \in C : \deg(v) > 0$  do
5:      $H \leftarrow G - E(C)$ 
6:      $v \leftarrow$  any vertex in  $V(C)$  such that  $\deg(v) > 0$ 
7:      $C' \leftarrow$  any circuit  $v \sim v$  in  $H$ 
8:      $C \leftarrow C \otimes C'$  ▷ Glue  $C' = v \sim v$  with  $C$  via  $v$ 
9:   return  $C$ 

```



Q : Time Complexity?

Q : Data Structures?

$O(m)$: Using doubly linked list

Fleury's Algorithm (1883)

(I) $v_0 \in V(G)$; $P_0 = v_0$

(II) Suppose $C_i = v_0 e_1 v_1 e_2 \cdots e_i v_i$.

Choose e_{i+1} from $E(G) - \{e_1, e_2, \cdots, e_i\}$ as follows:

- (a) e_{i+1} is incident with v_i
- (b) Unless there is no alternative, e_{i+1} is not a bridge of $G - \{e_1, e_2, \cdots, e_i\}$

(III) Stop when step (II) can no longer be implemented

Fleury's Algorithm (1883)

(I) $v_0 \in V(G)$; $P_0 = v_0$

(II) Suppose $C_i = v_0 e_1 v_1 e_2 \cdots e_i v_i$.

Choose e_{i+1} from $E(G) - \{e_1, e_2, \cdots, e_i\}$ as follows:

- (a) e_{i+1} is incident with v_i
- (b) Unless there is no alternative, e_{i+1} is not a bridge of $G - \{e_1, e_2, \cdots, e_i\}$

(III) Stop when step (II) can no longer be implemented



Fleury's Algorithm (1883)

(I) $v_0 \in V(G)$; $P_0 = v_0$

(II) Suppose $C_i = v_0 e_1 v_1 e_2 \cdots e_i v_i$.

Choose e_{i+1} from $E(G) - \{e_1, e_2, \cdots, e_i\}$ as follows:

- (a) e_{i+1} is incident with v_i
- (b) Unless there is no alternative, e_{i+1} is not a bridge of $G - \{e_1, e_2, \cdots, e_i\}$

(III) Stop when step (II) can no longer be implemented



Fleury's Algorithm (1883)

(I) $v_0 \in V(G)$; $P_0 = v_0$

(II) Suppose $C_i = v_0 e_1 v_1 e_2 \cdots e_i v_i$.

Choose e_{i+1} from $E(G) - \{e_1, e_2, \cdots, e_i\}$ as follows:

- (a) e_{i+1} is incident with v_i
- (b) Unless there is no alternative, e_{i+1} is not a bridge of $G - \{e_1, e_2, \cdots, e_i\}$

(III) Stop when step (II) can no longer be implemented



Fleury's Algorithm (1883)

(I) $v_0 \in V(G)$; $P_0 = v_0$

(II) Suppose $C_i = v_0 e_1 v_1 e_2 \cdots e_i v_i$.

Choose e_{i+1} from $E(G) - \{e_1, e_2, \cdots, e_i\}$ as follows:

- (a) e_{i+1} is incident with v_i
- (b) Unless there is no alternative, e_{i+1} is not a bridge of $G - \{e_1, e_2, \cdots, e_i\}$

(III) Stop when step (II) can no longer be implemented



PROOF

Theorem (Bridges in Fleury's Algorithm)

At any stage, v_i is incident with ≤ 1 bridge in $E(G) - \{e_1, e_2, \dots, e_i\}$.

Theorem (Bridges in Fleury's Algorithm)

At any stage, v_i is incident with ≤ 1 bridge in $E(G) - \{e_1, e_2, \dots, e_i\}$.

By Contradiction.

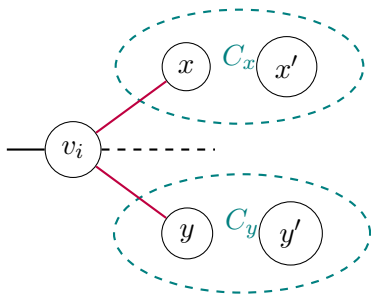
Suppose that v_i is incident with ≥ 2 bridges in $E(G) - \{e_1, e_2, \dots, e_i\}$.

Theorem (Bridges in Fleury's Algorithm)

At any stage, v_i is incident with ≤ 1 bridge in $E(G) - \{e_1, e_2, \dots, e_i\}$.

By Contradiction.

Suppose that v_i is incident with ≥ 2 bridges in $E(G) - \{e_1, e_2, \dots, e_i\}$.

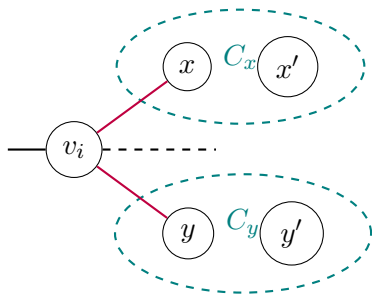


Theorem (Bridges in Fleury's Algorithm)

At any stage, v_i is incident with ≤ 1 bridge in $E(G) - \{e_1, e_2, \dots, e_i\}$.

By Contradiction.

Suppose that v_i is incident with ≥ 2 bridges in $E(G) - \{e_1, e_2, \dots, e_i\}$.



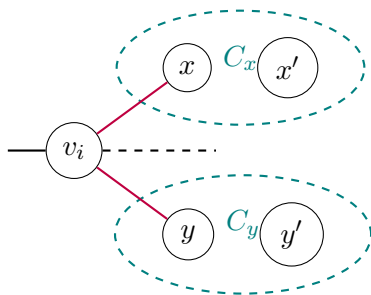
$$\exists x' \in C_x : \deg(x) \text{ is odd}$$

Theorem (Bridges in Fleury's Algorithm)

At any stage, v_i is incident with ≤ 1 bridge in $E(G) - \{e_1, e_2, \dots, e_i\}$.

By Contradiction.

Suppose that v_i is incident with ≥ 2 bridges in $E(G) - \{e_1, e_2, \dots, e_i\}$.



$\exists x' \in C_x : \deg(x) \text{ is odd}$

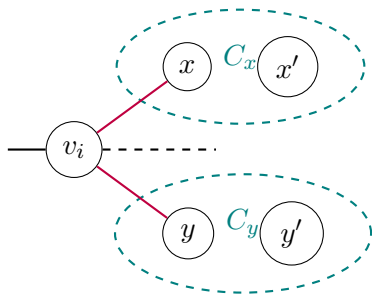
$\exists y' \in C_y : \deg(y) \text{ is odd}$

Theorem (Bridges in Fleury's Algorithm)

At any stage, v_i is incident with ≤ 1 bridge in $E(G) - \{e_1, e_2, \dots, e_i\}$.

By Contradiction.

Suppose that v_i is incident with ≥ 2 bridges in $E(G) - \{e_1, e_2, \dots, e_i\}$.



$\exists x' \in C_x : \deg(x) \text{ is odd}$

$\exists y' \in C_y : \deg(y) \text{ is odd}$

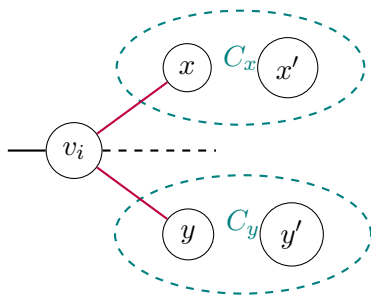
We have found 2 odd vertices.

Theorem (Bridges in Fleury's Algorithm)

At any stage, v_i is incident with ≤ 1 bridge in $E(G) - \{e_1, e_2, \dots, e_i\}$.

By Contradiction.

Suppose that v_i is incident with ≥ 2 bridges in $E(G) - \{e_1, e_2, \dots, e_i\}$.



$\exists x' \in C_x : \deg(x)$ is odd

$\exists y' \in C_y : \deg(y)$ is odd

We have found 2 odd vertices.

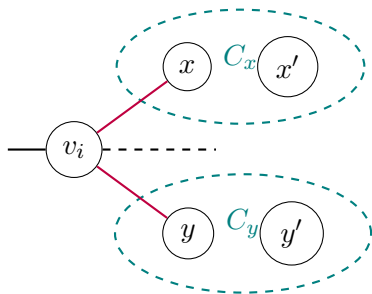
Q : What is the contradiction?

Theorem (Bridges in Fleury's Algorithm)

At any stage, v_i is incident with ≤ 1 bridge in $E(G) - \{e_1, e_2, \dots, e_i\}$.

By Contradiction.

Suppose that v_i is incident with ≥ 2 bridges in $E(G) - \{e_1, e_2, \dots, e_i\}$.



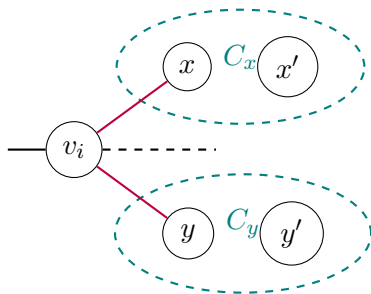
$\exists x' \in C_x : \deg(x) \text{ is odd}$

$\exists y' \in C_y : \deg(y) \text{ is odd}$

We have found 2 odd vertices.

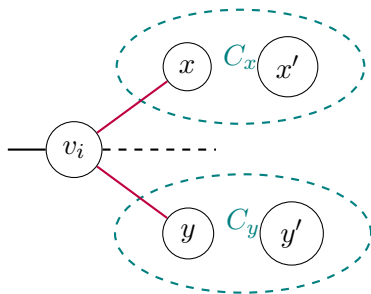
Q : What is the contradiction?

Is $\deg(v_i)$ odd or even?



We have found 2 odd vertices.

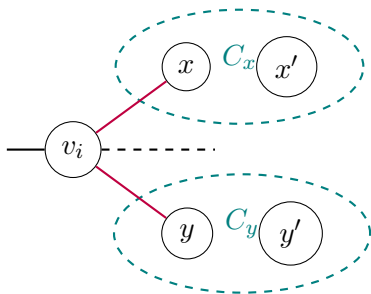
Is $\deg(v_i)$ odd or even?



We have found 2 odd vertices.

Is $\deg(v_i)$ odd or even?

CASE I: $\deg(v_i)$ is odd.

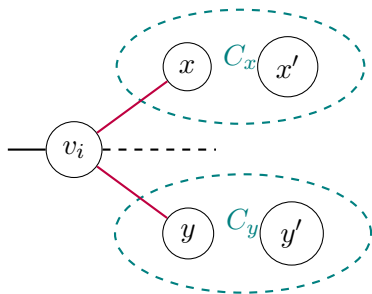


We have found 2 odd vertices.

Is $\deg(v_i)$ odd or even?

CASE I: $\deg(v_i)$ is odd.

Contradiction:



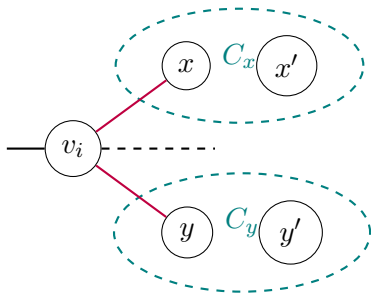
We have found 2 odd vertices.

Is $\deg(v_i)$ odd or even?

CASE I: $\deg(v_i)$ is odd.

Contradiction:

Only v_0 and v_i can have odd degrees!



We have found 2 odd vertices.

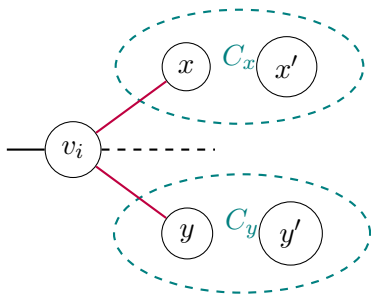
Is $\deg(v_i)$ odd or even?

CASE I: $\deg(v_i)$ is odd.

CASE II: $\deg(v_i)$ is even.

Contradiction:

Only v_0 and v_i can have odd degrees!



We have found 2 odd vertices.

Is $\deg(v_i)$ odd or even?

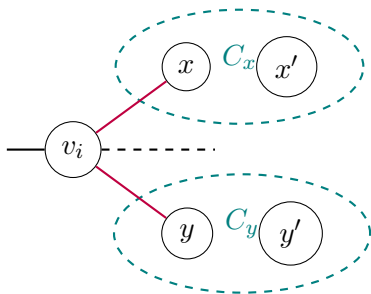
CASE I: $\deg(v_i)$ is odd.

Contradiction:

Only v_0 and v_i can have odd degrees!

CASE II: $\deg(v_i)$ is even.

$$v_i = v_0$$



We have found 2 odd vertices.

Is $\deg(v_i)$ odd or even?

CASE I: $\deg(v_i)$ is odd.

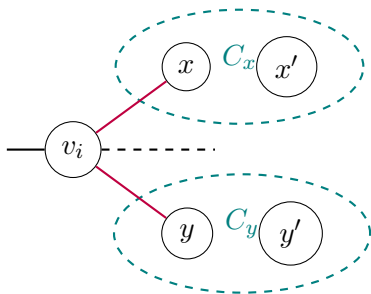
Contradiction:

Only v_0 and v_i can have odd degrees!

CASE II: $\deg(v_i)$ is even.

$$v_i = v_0$$

Contradiction:



We have found 2 odd vertices.

Is $\deg(v_i)$ odd or even?

CASE I: $\deg(v_i)$ is odd.

Contradiction:

Only v_0 and v_i can have odd degrees!

CASE II: $\deg(v_i)$ is even.

$$v_i = v_0$$

Contradiction: No odd vertices!

1: **procedure** FLEURY(G)

2: $v_0 \in V(G)$

3: $C \leftarrow v_0$

4: $i \leftarrow 0$, $V_0 \leftarrow V(G)$, $E_0 \leftarrow E(G)$

▷ Choose any starting vertex

▷ Keep track of the circuit

```
1: procedure FLEURY( $G$ )
2:    $v_0 \in V(G)$ 
3:    $C \leftarrow v_0$ 
4:    $i \leftarrow 0$ ,  $V_0 \leftarrow V(G)$ ,  $E_0 \leftarrow E(G)$ 
5:   while  $\deg(v_i) > 0$  in  $E_i$  do
```

▷ Choose any starting vertex
▷ Keep track of the circuit
▷ Stop otherwise

```
15: return  $C$ 
```

```

1: procedure FLEURY( $G$ )
2:    $v_0 \in V(G)$                                 ▷ Choose any starting vertex
3:    $C \leftarrow v_0$                                ▷ Keep track of the circuit
4:    $i \leftarrow 0$ ,  $V_0 \leftarrow V(G)$ ,  $E_0 \leftarrow E(G)$ 

5:   while  $\deg(v_i) > 0$  in  $E_i$  do                ▷ Stop otherwise
6:     if  $\deg(v_i) = 1$  in  $E_i$  then                ▷ No alternative: go the bridge
7:        $e_{i+1} \triangleq v_i v_{i+1}$ 
8:                                     ▷ Delete the isolated vertex  $v_i$ 
9:     else                                          ▷ Have alternatives: don't go the bridge
10:      Choose  $e_{i+1} \triangleq v_i v_{i+1}$  that is not a bridge of  $(V_i, E_i)$ 
11:                                     ▷ No isolated vertex produced

15:  return  $C$ 

```

```

1: procedure FLEURY( $G$ )
2:    $v_0 \in V(G)$                                 ▷ Choose any starting vertex
3:    $C \leftarrow v_0$                                ▷ Keep track of the circuit
4:    $i \leftarrow 0$ ,  $V_0 \leftarrow V(G)$ ,  $E_0 \leftarrow E(G)$ 

5:   while  $\deg(v_i) > 0$  in  $E_i$  do                ▷ Stop otherwise
6:     if  $\deg(v_i) = 1$  in  $E_i$  then                ▷ No alternative: go the bridge
7:        $e_{i+1} \triangleq v_i v_{i+1}$ 
8:                                     ▷ Delete the isolated vertex  $v_i$ 
9:     else                                          ▷ Have alternatives: don't go the bridge
10:      Choose  $e_{i+1} \triangleq v_i v_{i+1}$  that is not a bridge of  $(V_i, E_i)$ 
11:                                     ▷ No isolated vertex produced

12:       $C \leftarrow C e_{i+1} v_{i+1}$ 
13:       $E_{i+1} \leftarrow E_i - \{e_{i+1}\}$ 
14:       $i \leftarrow i + 1$ 

15:   return  $C$ 

```

```

1: procedure FLEURY( $G$ )
2:    $v_0 \in V(G)$                                 ▷ Choose any starting vertex
3:    $C \leftarrow v_0$                                 ▷ Keep track of the circuit
4:    $i \leftarrow 0, V_0 \leftarrow V(G), E_0 \leftarrow E(G)$ 

5:   while  $\deg(v_i) > 0$  in  $E_i$  do                                ▷ Stop otherwise
6:     if  $\deg(v_i) = 1$  in  $E_i$  then                                ▷ No alternative: go the bridge
7:        $e_{i+1} \triangleq v_i v_{i+1}$ 
8:        $V_{i+1} \leftarrow V_i - \{v_i\}$                                 ▷ Delete the isolated vertex  $v_i$ 
9:     else                                ▷ Have alternatives: don't go the bridge
10:      Choose  $e_{i+1} \triangleq v_i v_{i+1}$  that is not a bridge of  $(V_i, E_i)$ 
11:       $V_{i+1} \leftarrow V_i$                                 ▷ No isolated vertex produced

12:       $C \leftarrow C e_{i+1} v_{i+1}$ 
13:       $E_{i+1} \leftarrow E_i - \{e_{i+1}\}$ 
14:       $i \leftarrow i + 1$ 

15:   return  $C$ 

```

PROOF

We need to prove that

PROOF

We need to prove that C returned by FLEURY is an Eulerian circuit.

PROOF

We need to prove that C returned by FLEURY is an Eulerian circuit.

Definition (Eulerian Circuit)

A connected graph is **Eulerian** if there exists a **closed trail** that **includes every edge of G** .

A **trail** is a walk in which all the edges are distinct.

PROOF

We need to prove that C returned by FLEURY is an Eulerian circuit.

Definition (Eulerian Circuit)

A connected graph is **Eulerian** if there exists a **closed trail** that **includes every edge of G** .

A **trail** is a walk in which all the edges are distinct.

Closed

Trail

Include every edge of G

PROOF

We need to prove that C returned by FLEURY is an Eulerian circuit.

Definition (Eulerian Circuit)

A connected graph is **Eulerian** if there exists a **closed trail** that **includes every edge of G** .

A **trail** is a walk in which all the edges are distinct.

Closed

Trail

Include every edge of G

\therefore even degrees

PROOF

We need to prove that C returned by FLEURY is an Eulerian circuit.

Definition (Eulerian Circuit)

A connected graph is **Eulerian** if there exists a **closed trail** that **includes every edge of G** .

A **trail** is a walk in which all the edges are distinct.

Closed

Trail

Include every edge of G

\therefore even degrees \therefore used edges are deleted

PROOF

We need to prove that C returned by FLEURY is an Eulerian circuit.

Definition (Eulerian Circuit)

A connected graph is **Eulerian** if there exists a **closed trail** that **includes every edge of G** .

A **trail** is a walk in which all the edges are distinct.

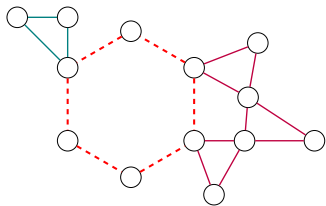
Closed

Trail

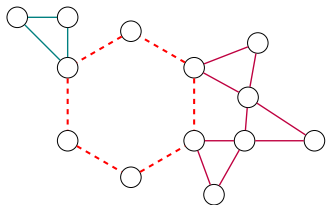
Include every edge of G

\because even degrees \because used edges are deleted

By Contradiction.

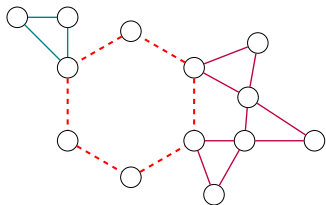


Include every edge of G
By Contradiction.



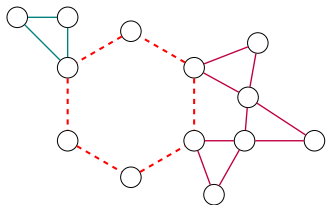
Include every edge of G
By Contradiction.

We know that $C : v_0 \sim v_0$



Include every edge of G
By Contradiction.

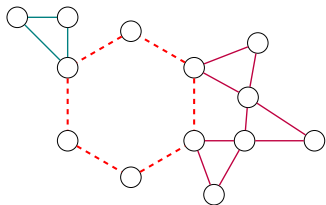
We know that $C : v_0 \sim v_0$ $E' \triangleq E(G) - E(C) \neq \emptyset$



Include every edge of G
By Contradiction.

We know that $C : v_0 \sim v_0$ $E' \triangleq E(G) - E(C) \neq \emptyset$

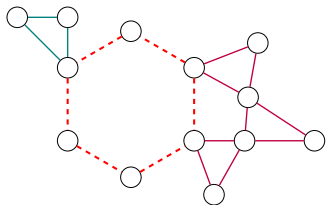
$$\deg(v_0) = 0$$



Include every edge of G
By Contradiction.

We know that $C : v_0 \sim v_0 \quad E' \triangleq E(G) - E(C) \neq \emptyset$

$\deg(v_0) = 0$ (Otherwise, FLEURY is not terminated.)

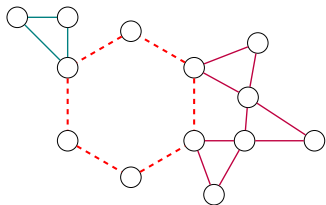


Include every edge of G
By Contradiction.

We know that $C : v_0 \sim v_0$ $E' \triangleq E(G) - E(C) \neq \emptyset$

$\deg(v_0) = 0$ (Otherwise, FLEURY is not terminated.)

$G|_{E'}$ is disconnected from v_0



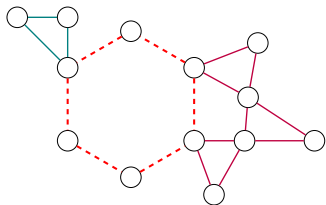
Include every edge of G
By Contradiction.

We know that $C : v_0 \sim v_0$ $E' \triangleq E(G) - E(C) \neq \emptyset$

$\deg(v_0) = 0$ (Otherwise, FLEURY is not terminated.)

$G|_{E'}$ is disconnected from v_0

Impossible:



Include every edge of G
By Contradiction.

We know that $C : v_0 \sim v_0$ $E' \triangleq E(G) - E(C) \neq \emptyset$

$\deg(v_0) = 0$ (Otherwise, FLEURY is not terminated.)

$G|_{E'}$ is disconnected from v_0

Impossible:

- (I) Don't go the bridge unless there is no alternative.
- (II) Delete the isolated vertex left by going the bridge.





Office 302

Mailbox: H016

hfwei@nju.edu.cn