



# グラフの探索2

## EULER閉路とHAMILTON閉路

# グラフ探索

- Euler閉路の探索
  - 全ての弧を一度ずつ経由する閉路
  - 「一筆描き」
- Hamilton閉路の探索
  - 全ての頂点を一度ずつ経由する閉路
- 閉路(circle, closed path)
  - 始点と終点の一致する道
- 一つだけ見つけるのではなく、列挙する

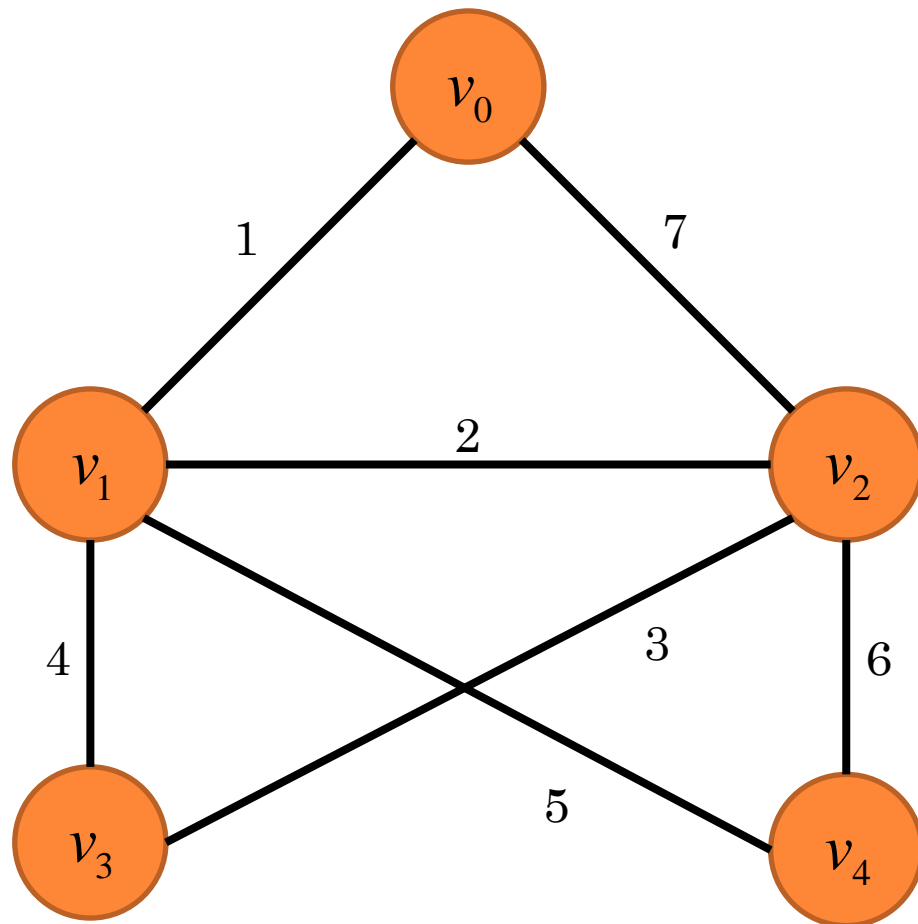


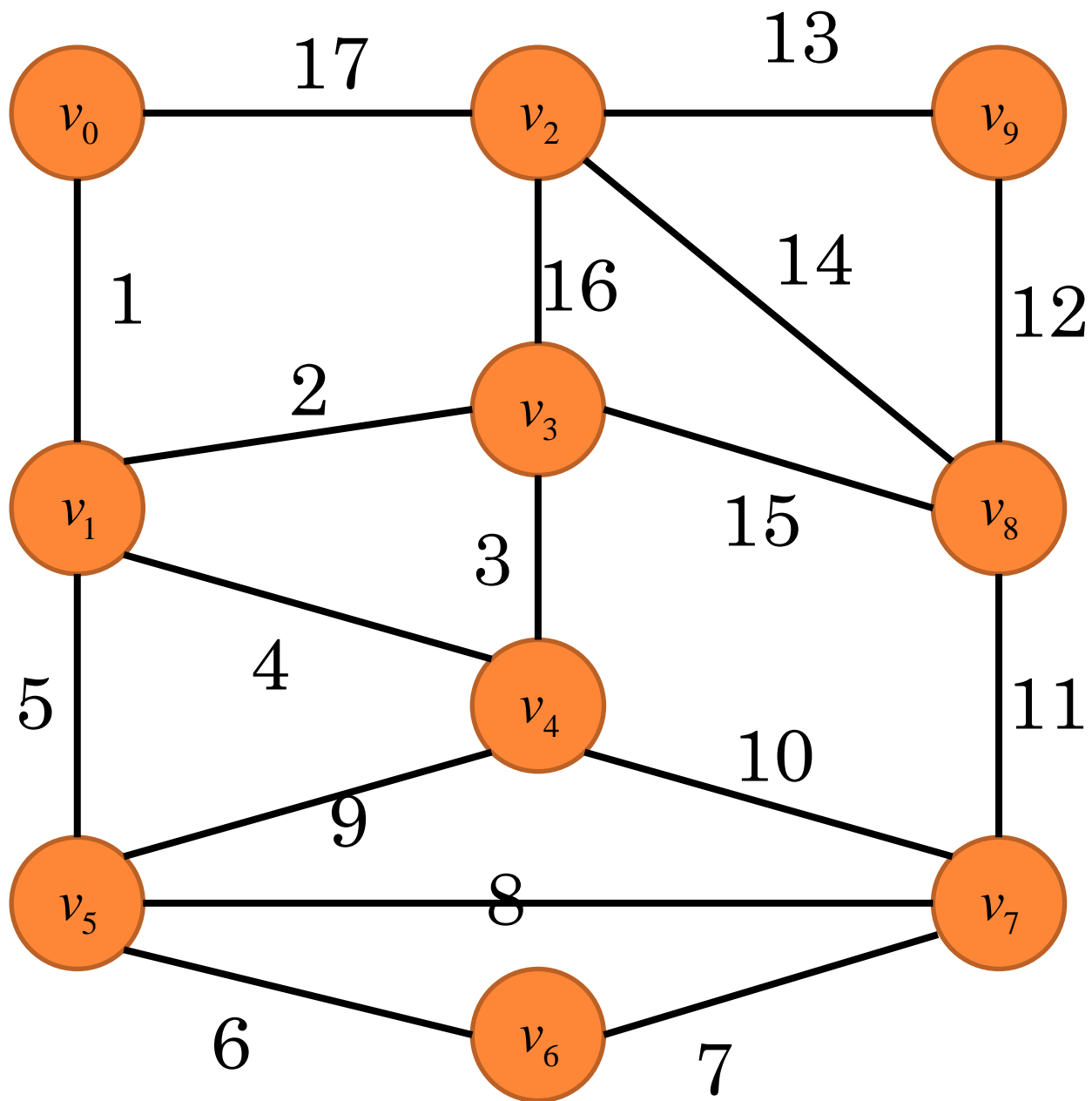
# EULER閉路(EULER CIRCLE)

- 一筆書き
- 無向グラフが対象
- 全ての弧を一度ずつ経由して始点に戻る道
- 全ての点の次数が偶数
  - 次数が奇数だったらどうなる？



# 簡単なEULER閉路の例





## EULER閉路の列挙の方針

- 使用した弧の列を管理する
- 深さ優先で、一つの閉路を見つける
  - 使った弧の一覧を保持
- 分岐点まで戻って、他の閉路を見つける
  - 戻るたびに、対応する弧を一覧から削除
- 上記を繰り返す



# EULER閉路の列挙アルゴリズム

```
search(v){
  if ((v == r) ^ (|A| == |AEuler|)) {
    見つけた Euler 閉路を保存
  } else {
    forall (a ∈ δv) {
      if (a ∉ AEuler) {
        AEuler に a をアペンド
        w は a の反対の終端
        search(w)
        AEuler から a を削除
      }
    }
  }
}
```

$A_{\text{Euler}}$  : 既に経由した弧の列

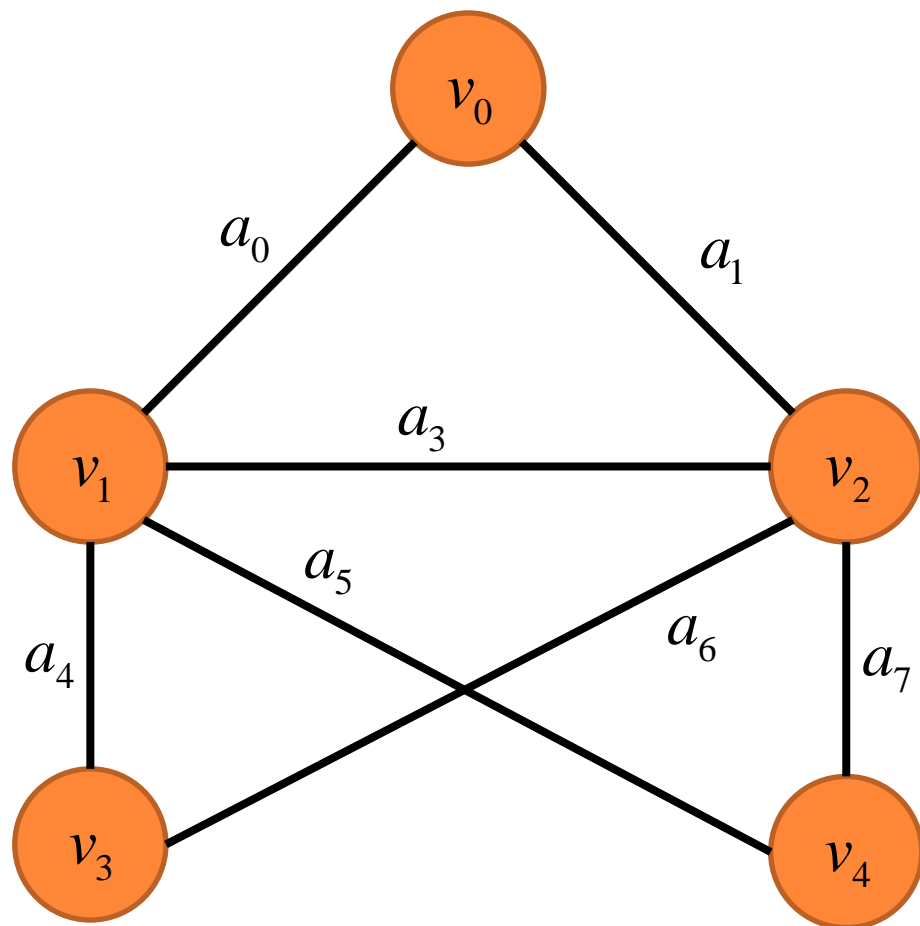
初期値は  $A_{\text{Euler}} = \emptyset$

$r$  : 始点

$|A|$  : 弧の数



# 列举の例



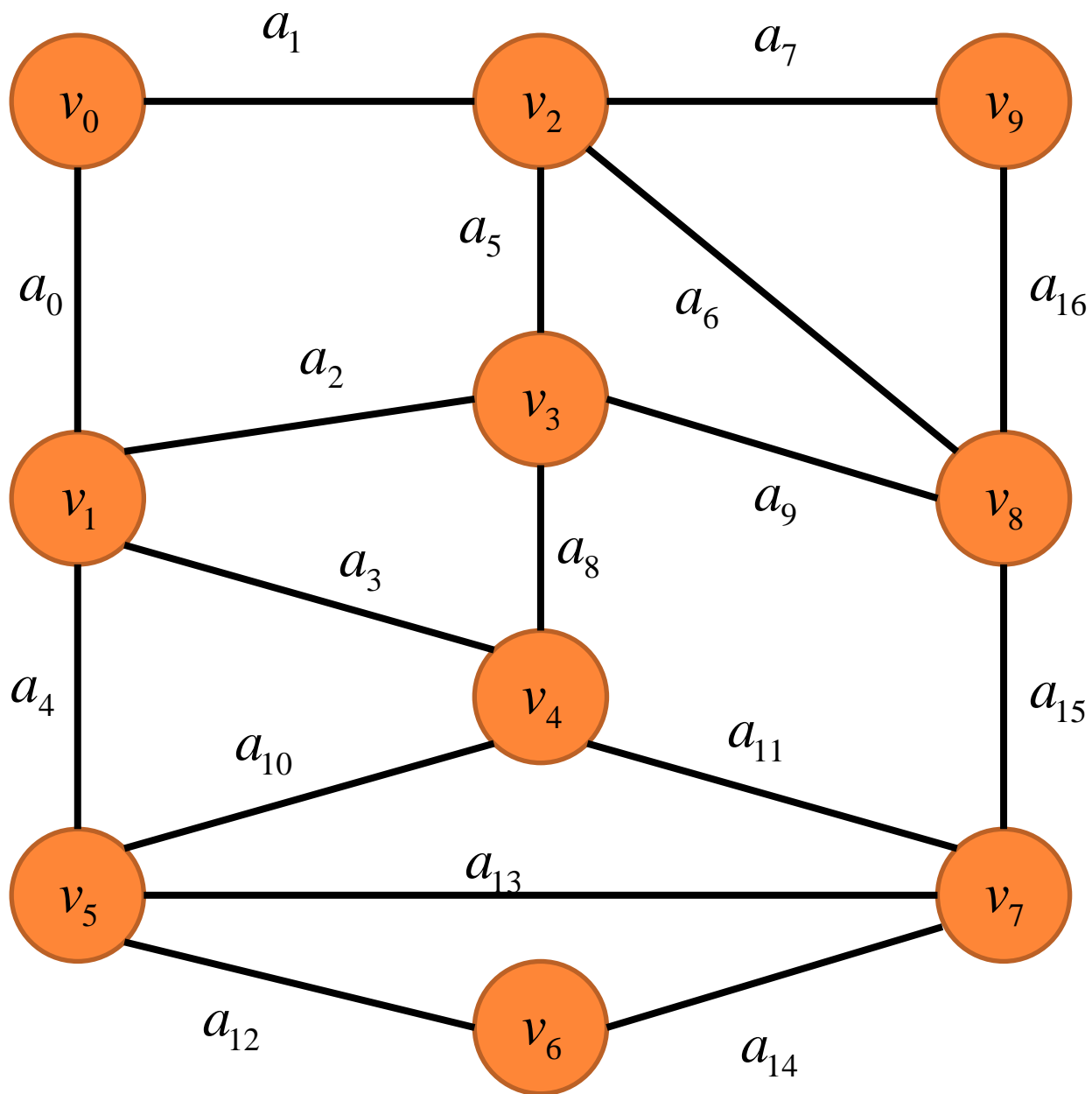


# 探索の経過

$a_0$	$a_3$	$a_1$	×				
		$a_6$	$a_4$	$a_5$	$a_7$	$a_1$	○
		$a_7$	$a_5$	$a_4$	$a_6$	$a_1$	○
	$a_4$	$a_6$	$a_1$	×			
			$a_3$	$a_5$	$a_7$	$a_1$	○
			$a_7$	$a_5$	$a_3$	$a_1$	○
	$a_5$	$a_7$	$a_3$	$a_4$	$a_6$	$a_1$	○
			$a_6$	$a_4$	$a_3$	$a_1$	○

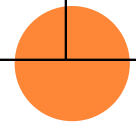
以下省略





$a_0$	$a_2$	$a_5$	$a_1$	×												
			$a_6$	$a_9$	$a_8$	$a_3$	$a_4$	$a_{10}$	$a_{11}$	$a_{13}$	$a_{12}$	$a_{14}$	$a_{15}$	$a_{16}$	$a_7$	$a_1$
								$a_{12}$	$a_{14}$	$a_{13}$	$a_{10}$	$a_{11}$	$a_{15}$	$a_{16}$	$a_7$	$a_1$
						$a_{10}$	$a_4$	$a_3$	$a_{11}$	$a_{13}$	$a_{12}$	$a_{14}$	$a_{15}$	$a_{16}$	$a_7$	$a_1$
							$a_{12}$	$a_{14}$	$a_{11}$	$a_3$	$a_4$	$a_{13}$	$a_{15}$	$a_{16}$	$a_7$	$a_1$
								$a_{13}$	$a_4$	$a_3$	$a_{11}$	$a_{15}$	$a_{16}$	$a_7$	$a_1$	
								$a_{15}$	$a_{16}$	$a_7$	$a_1$	×				
							$a_{13}$	$a_{11}$	$a_3$	$a_4$	$a_{12}$	$a_{14}$	$a_{15}$	$a_{16}$	$a_7$	$a_1$
								$a_{14}$	$a_{12}$	$a_4$	$a_3$	$a_{11}$	$a_{15}$	$a_{16}$	$a_7$	$a_1$
						$a_{11}$	$a_{13}$	$a_4$	$a_3$	$a_{10}$	$a_{12}$	$a_{14}$	$a_{15}$	$a_{16}$	$a_7$	$a_1$
								$a_{10}$	$a_3$	$a_4$	$a_{12}$	$a_{14}$	$a_{15}$	$a_{16}$	$a_7$	$a_1$
								$a_{12}$	$a_{14}$	$a_{15}$	$a_{16}$	$a_7$	$a_1$	×		
							$a_{14}$	$a_{12}$	$a_4$	$a_3$	$a_{10}$	$a_{13}$	$a_{15}$	$a_{14}$	$a_{12}$	$a_4$
								$a_{10}$	$a_3$	$a_4$	$a_{13}$	$a_{15}$	$a_{14}$	$a_{12}$	$a_4$	
								$a_{13}$	$a_{15}$	$a_{16}$	$a_7$	$a_1$	×			
							$a_{15}$	$a_{16}$	$a_7$	$a_1$	×					

以下省略



# HAMILTON 閉路

- 無向グラフが対象
- 全ての点を一度ずつ経由して始点に戻る道
- 巡回セールスマン問題の厳密解を得る際に必要

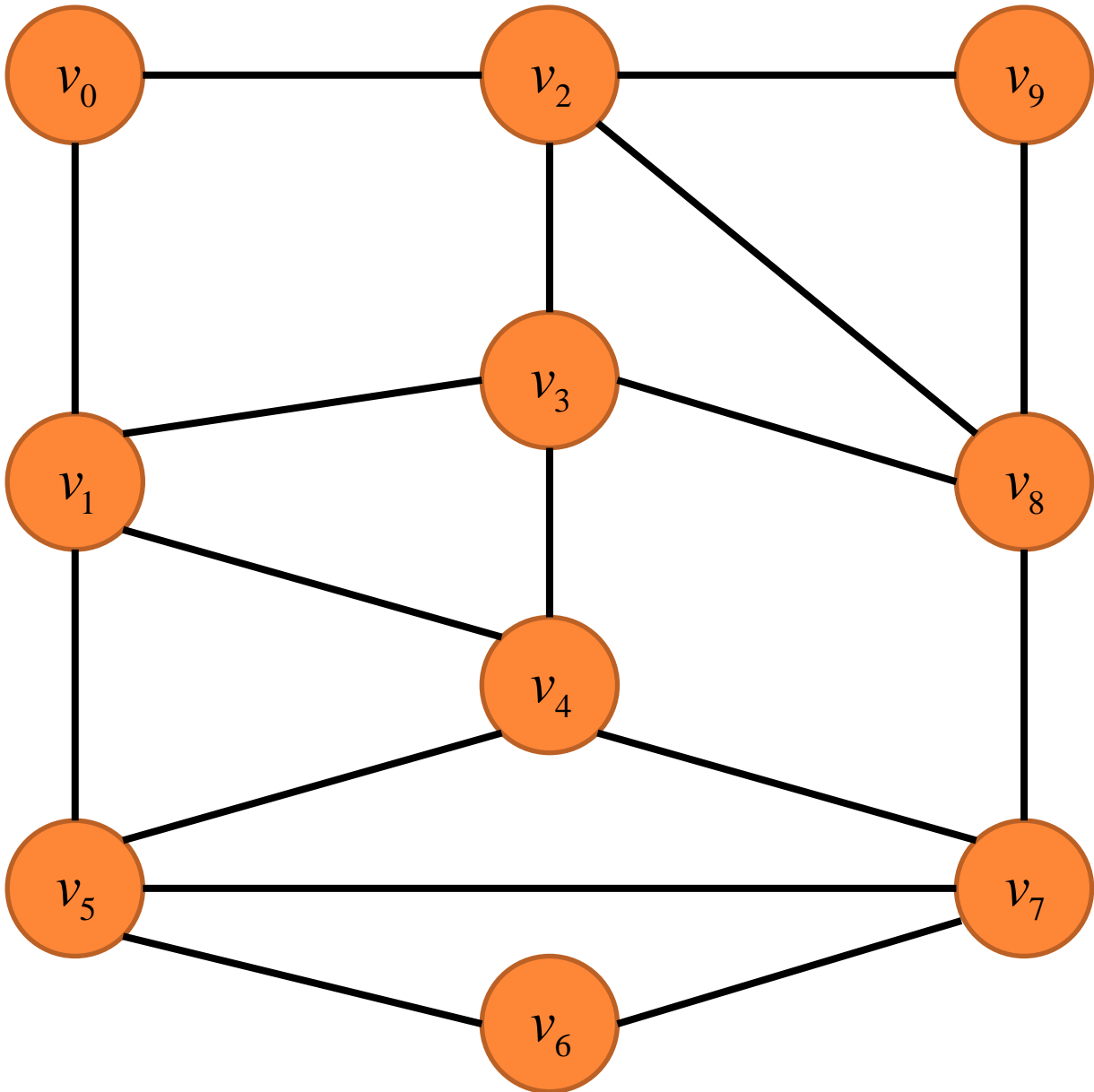


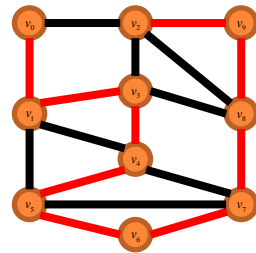
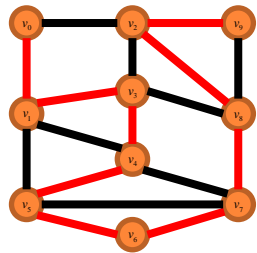
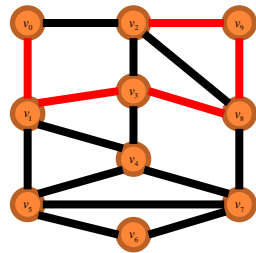
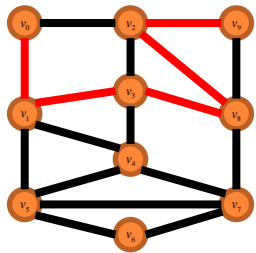
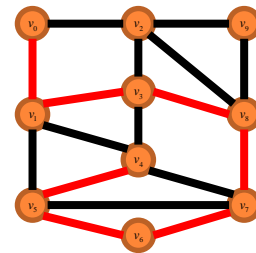
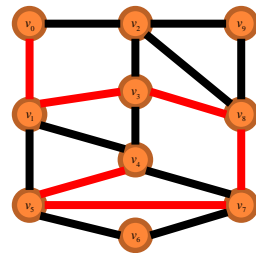
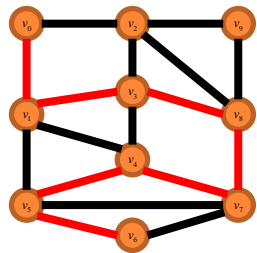
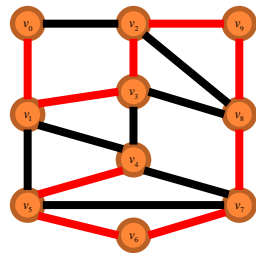
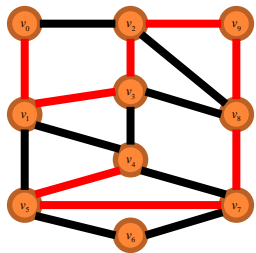
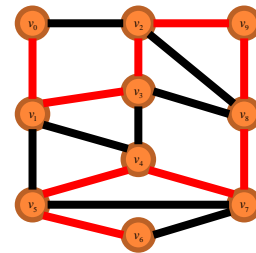
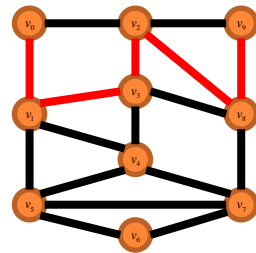
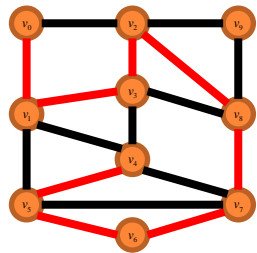
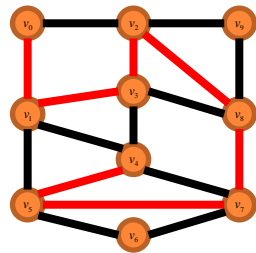
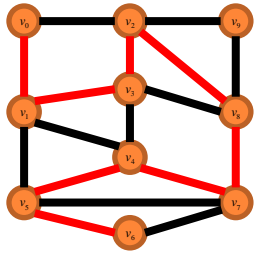
# HAMILTON閉路の列挙アルゴリズム

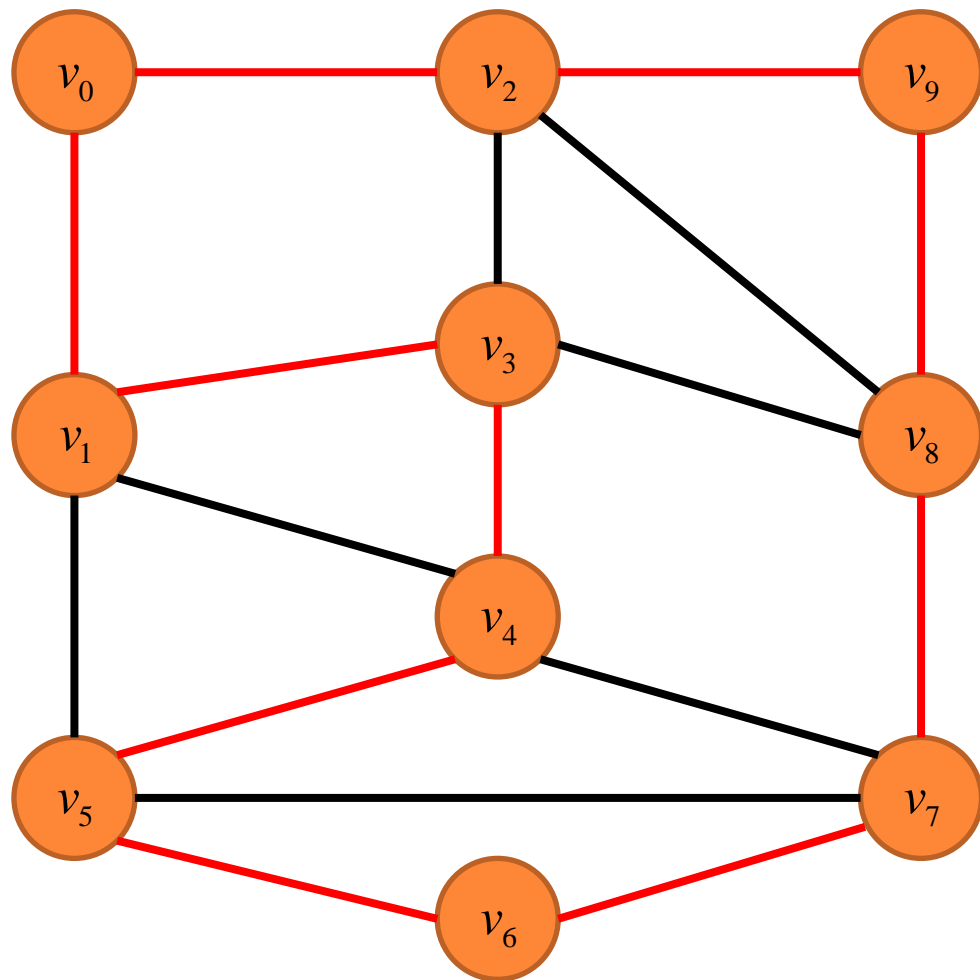
```
search (v) {  
  forall (a ∈ δv) {  
    wはaの反対の終端  
    if ((w == r) ∧ (|L| == |V|)) { 見つけた Hamilton 閉路を保存 }  
    else {  
      if (w ∉ L) {  
        Lにwを追加  
        search (w)  
        Lからwを削除  
      }  
    }  
  }  
}
```

$L$  : 既に経由した点の集合  
初期値 :  $L = \emptyset$   
 $r$  : 始点

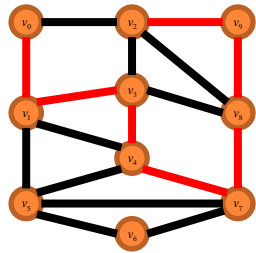
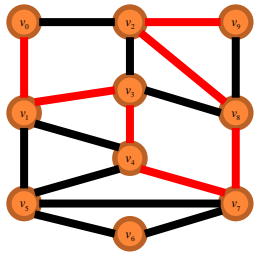
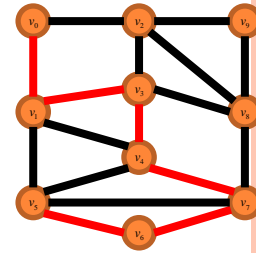
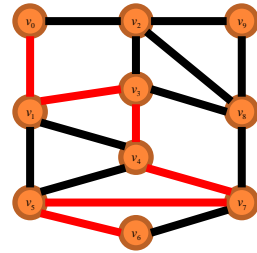
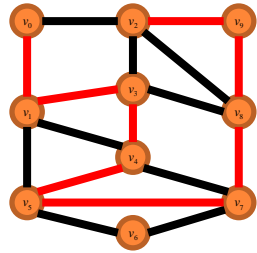
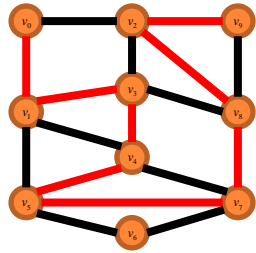
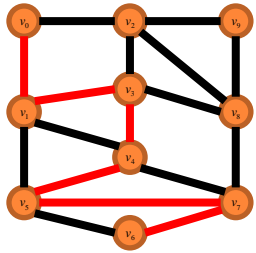




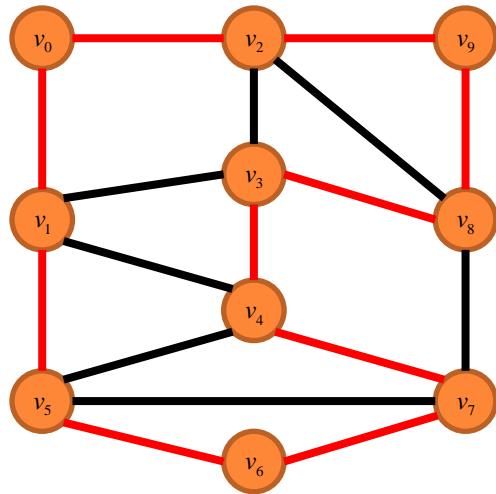








...

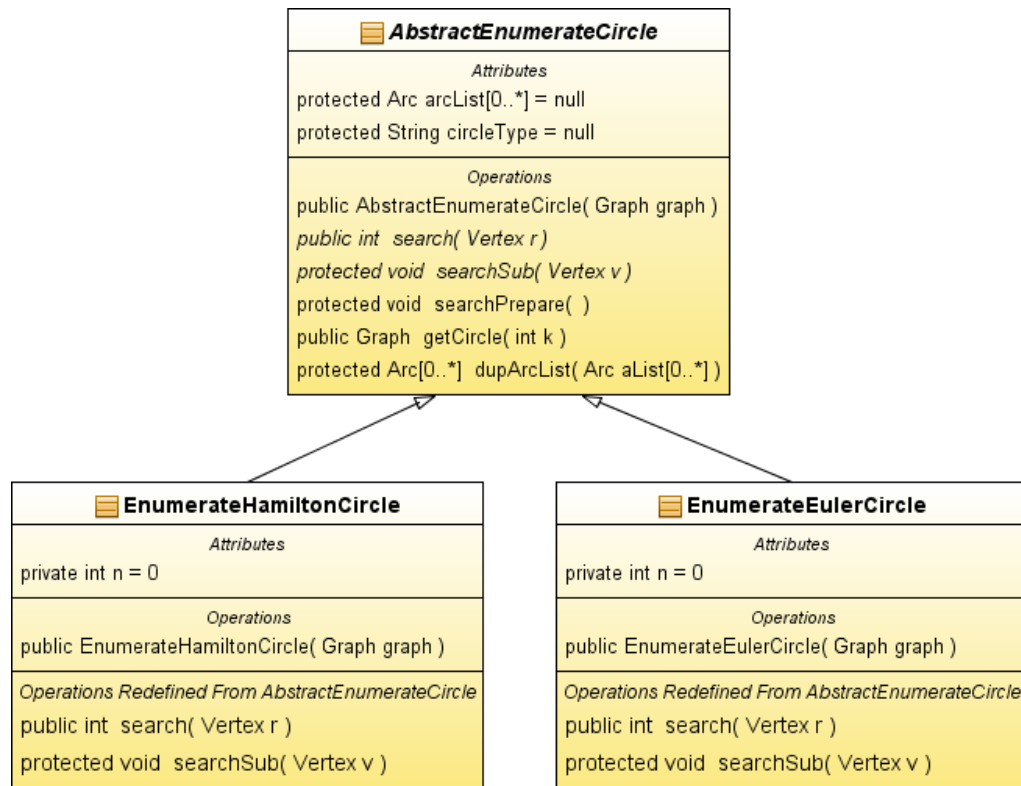


# JAVAプログラムへ

- アブストラクトクラス
  - AbstractEnumerateCircle
- Euler閉路の列挙
  - EnumerateEulerCircle
- Hamilton閉路の列挙
  - EnumerateHamiltonCircle



# クラスの関係



## ○ メソッドsearch

- 始点を指定して、探索を準備する。
- 探索結果(circle)を保存する準備
- circleに対応する頂点または弧のリストを準備
- 再帰的メソッドsearchSubを呼び出す
- 発見したcircleの数を返す



- メソッドsearchSub

- 再帰的メソッド

- メソッドgetCircle

- k番目のcircleを返す
- 指定されたcircleに対応する頂点または弧のリストからグラフを構成する



