「グラフと組み合わせ」試験問題
–2015年度前期–

2015/8/3

解答例

1 数学的帰納法

自然数 $n \geq 1$ に対する以下の公式を、数学的帰納法を用いて証明しなさい。

$$\sum_{k=1}^{n} k(k-1) = \frac{1}{3}(n-1)n(n+1) \tag{1}$$

解答例

1. $n=1$ の場合、左辺は
$$\sum_{k=1}^{1} k(k-1) = 1 \times 0 = 0$$
右辺は
$$\frac{1}{3}(n-1)n(n+1) = \frac{1}{3} \times 0 \times 1 \times 2 = 0$$
となり、等式が成り立つ。

2. ある n について、式 (1) が成り立つと仮定する。$n+1$ の場合について考える。
$$\begin{aligned}\sum_{k=1}^{n+1} k(k-1) &= \sum_{k=1}^{n} k(k-1) + n(n+1) \\ &= \frac{1}{3}(n-1)n(n+1) + n(n+1) \\ &= \frac{1}{3}n(n+1)(n+2)\end{aligned}$$
つまり、$n+1$ についての式 (1) となっている。以上より、数学的帰納法より、任意の自然数 n について式 (1) が成り立つ。

2 幅優先探索

有向グラフ $G = (V, A)$ に対する幅優先探索のアルゴリズムを Algorithm 1 に示す。探索の始点を $r \in V$ とする。$L \subseteq V$ は探索済みの頂点の集合である。また、Q は、経路が探索済みであり、その後探索の起点となる頂点を保存する待ち行列である。

```
L = ∅
Q = {r}
while(Q ≠ ∅){
    v = Q の先頭を取り出し
    forall(a ∈ δ⁺v){
        w = ∂⁻a
        if((w ∉ L) ∧ (w ∉ Q)){
            Q へ w を追加
        }
    }
    L ← L ∪ {v}
}
```

Algorithm 1: 幅優先探索。

2.1 探索の実行

Algorithm 1 の幅優先探索を図 1 のグラフに対して、v_0 を始点として行ないなさい。解答として、着目していた頂点 v、変更された集合 L 及び待ち行列 Q の while ループの最後における状態の変化として記述しなさい。

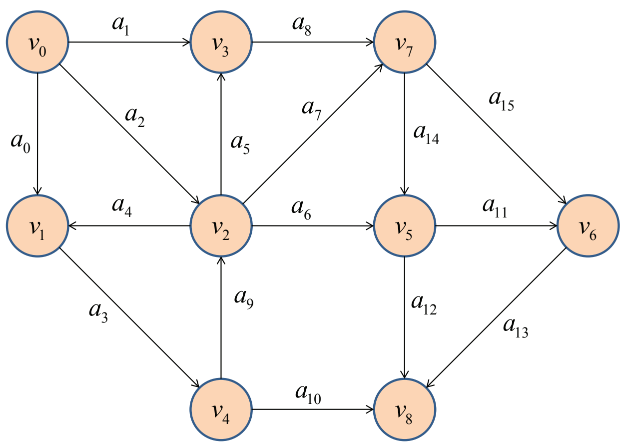


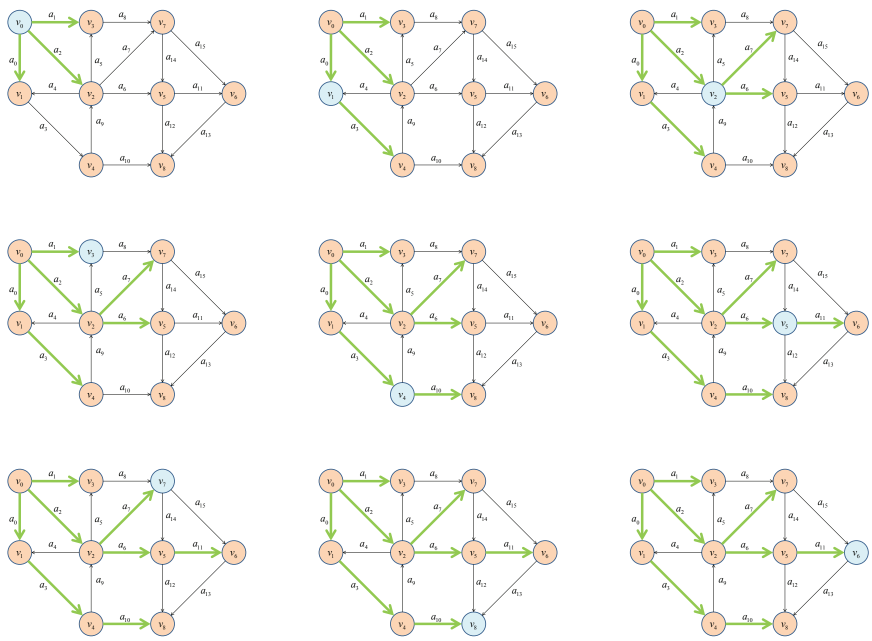
図 1: 幅優先探索を行う対象グラフ

解答例 `while` ループの最後における着目していた頂点 w、変更された集合 L 及び待ち行列 Q の変化を表 1 にまとめる。

表 1: `while` ループの最後における着目していた頂点 w、変更された集合 L 及び待ち行列 Q の変化

w	L	Q
	$\emptyset$	$[v_0]$
v_0	$\{v_0\}$	$[v_1, v_2, v_3]$
v_1	$\{v_0, v_1\}$	$[v_2, v_3, v_4]$
v_2	$\{v_0, v_1, v_2\}$	$[v_3, v_4, v_5, v_7]$
v_3	$\{v_0, v_1, v_2, v_3\}$	$[v_4, v_5, v_7]$
v_4	$\{v_0, v_1, v_2, v_3, v_4\}$	$[v_5, v_7, v_8]$
v_5	$\{v_0, v_1, v_2, v_3, v_4, v_5\}$	$[v_7, v_8, v_6]$
v_7	$\{v_0, v_1, v_2, v_3, v_4, v_5, v_7\}$	$[v_8, v_6]$
v_8	$\{v_0, v_1, v_2, v_3, v_4, v_5, v_7, v_8\}$	$[v_6]$
v_6	$\{v_0, v_1, v_2, v_3, v_4, v_5, v_6, v_7, v_8\}$	$[]$

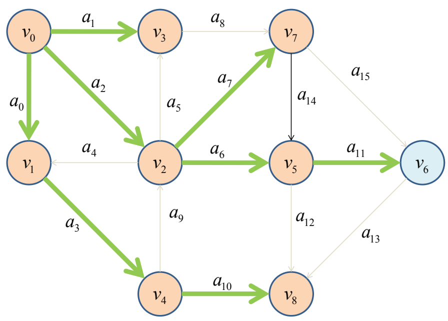
探索の様子を以下に図示する。



2.2 探索の結果

幅優先探索を実行して得られる極大木を示しなさい。

解答例



3　最小木

無向グラフ $G = (V, A)$ の各弧 $a \in A$ に重み $w : A \to R$ が定義されているとき、グラフ G のすべての頂点を接続する極大木のうち、重みの総和が最小になる木を最小木と呼ぶ。最小木を求めるアルゴリズムに Jarník-Prim のアルゴリズムがある。

$U \subseteq V$ をすでに極大木 $T \subseteq A$ の一部となった頂点の集合とする。探索の始点を $v_0 \in V$ とし、初期値は $U = \{v_0\}$、$T = \emptyset$ とすると、アルゴリズムは Algorithm 2 と記述できる。

$U = \{v_o\}$
$T = \emptyset$
while($U \neq V$){
　U と $V \backslash U$ を結ぶ弧のうち重み最小のものを a とする;
　a の $V \backslash U$ 側の頂点を w とする;
　$U = U \cup \{w\}$;//U に w を追加
　$T = T \cup \{a\}$;//T に a を追加
}

Algorithm 2: Jarník-Prim のアルゴリズム。//はコメント。

3.1　最小木を求める

Algorithm 2 を図 2 のグラフに適用し、探索の始点を v_0 として、最小木を求めることを考える。解答として集合 U に頂点が追加される順番を示しなさい。図 2 中の数字は各弧の重みを表す。

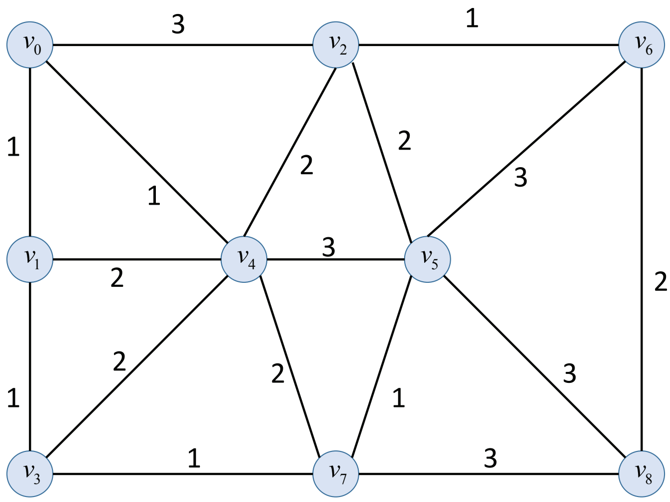
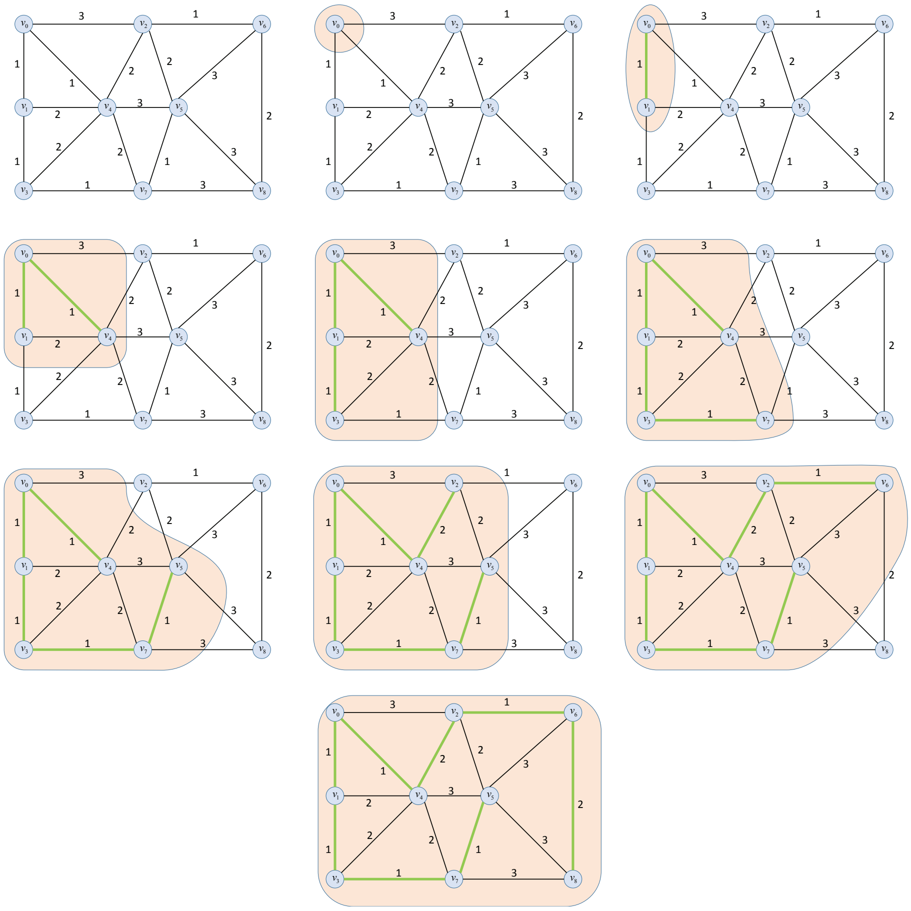


図 2: Algorithm 2 を用いる対象グラフ。各弧に添えられた数字はそれぞれの重みを表す。

解答例最小木が生成される様子を下図に示す。頂点の集合を囲む塗りつぶしが、U に相当する。U に頂点が追加される順番は以下のようになる。

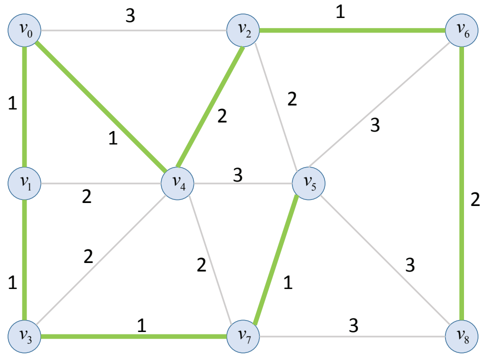
$$v_0 \to v_1 \to v_4 \to v_3 \to v_7 \to v_5 \to v_2 \to v_6 \to v_8$$



3.2　最小木

得られた最小木を示しなさい。

解答例



v_0
3
v_2
1
v_6
1
1
2
2
3
v_1
2
v_4
3
v_5
2
2
2
3
1
1
v_3
1
v_7
3
v_8