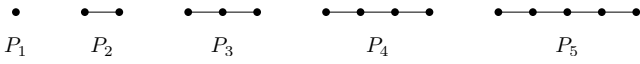


提出締切：2017年10月24日 講義終了時

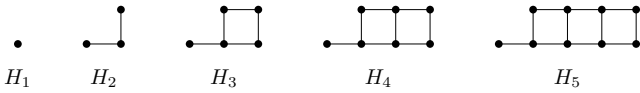
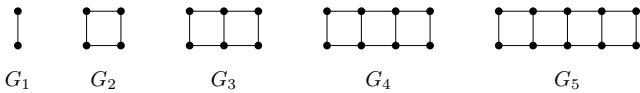
復習問題 2.1 自然数  $n \geq 1$  に対して、頂点数  $n$  の道を  $P_n$  と表記することにする。



グラフ  $P_n$  における独立集合の総数を  $a_n$  としたとき、次の漸化式が成り立つことを証明せよ。

$$a_n = \begin{cases} 2 & (n = 1 \text{ のとき}) \\ 3 & (n = 2 \text{ のとき}) \\ a_{n-1} + a_{n-2} & (n \geq 3 \text{ のとき}). \end{cases}$$

復習問題 2.2 自然数  $n \geq 1$  に対して、次の図で表されるグラフ  $G_n$  と  $H_n$  を考える。



グラフ  $G_n$  における独立集合の総数を  $b_n$  とし、グラフ  $H_n$  における独立集合の総数を  $c_n$  とする。次の漸化式が成り立つことを証明せよ。

$$b_n = \begin{cases} 3 & (n = 1 \text{ のとき}) \\ c_n + c_{n-1} & (n \geq 2 \text{ のとき}), \end{cases}$$

$$c_n = \begin{cases} 2 & (n = 1 \text{ のとき}) \\ b_{n-1} + c_{n-1} & (n \geq 2 \text{ のとき}). \end{cases}$$

復習問題 2.3 次のアルゴリズムを考える。

```
1: def fnct(n)
2:   print "a"
3:   if n > 2
4:     fnct(n-1)
5:     fnct(n-2)
6:   end
7: end
```

自然数  $n \geq 0$  に対して、 $\text{fnct}(n)$  を実行したときに出力される「a」の数を  $f_n$  とする。次の漸化式が成り立つことを証明せよ。

$$f_n = \begin{cases} 1 & (n \leq 2 \text{ のとき}) \\ 1 + f_{n-1} + f_{n-2} & (n \geq 3 \text{ のとき}). \end{cases}$$

復習問題 2.4 自然数  $a, b \geq 1$  に対して、 $a \geq b$  のとき、

$$a \bmod b \leq \left\lfloor \frac{a}{2} \right\rfloor$$

が成り立つことを証明せよ。ただし、 $a \bmod b$  は  $a$  を  $b$  で割った余りを表す。(注：問題 2.7 の結果を使ってもよい。)

復習問題 2.5 次のアルゴリズムを考える。

```
1: def gcd(a, b) # precondition: a >= b
2:   print "G"
3:   if b == 0
4:     return a
5:   else
6:     gcd(b, a % b)
7:   end
8: end
```

そして、任意の自然数  $n \geq 0$  に対して次の量を考える。

$$g_n = \max_{\substack{a \geq 1, \\ b \leq n}} \{\text{gcd}(a, b) \text{ の実行で出力される G の数} \}$$

このとき、次の式が成り立つことを証明せよ。

$$g_n = \begin{cases} 1 & (n = 0 \text{ のとき}) \\ \leq 2 + g_{\lfloor n/2 \rfloor} & (n \geq 1 \text{ のとき}). \end{cases}$$

(注：問題 2.4 の結果を使ってもよい。)

補足問題 2.6 自然数  $a, b \geq 1$  の最大公約数を  $\text{gcd}(a, b)$  で表す。任意の自然数  $a > b \geq 1$  に対して、 $\text{gcd}(a, b) = \text{gcd}(b, a \bmod b)$  が成り立つことを証明せよ。(注意：この事実から、ユークリッドのアルゴリズムの正当性が導かれる。)

補足問題 2.7 任意の自然数  $n \geq 0$  に対して、 $n - \lfloor n/2 \rfloor = \lfloor n/2 \rfloor$  が成り立つことを証明せよ。

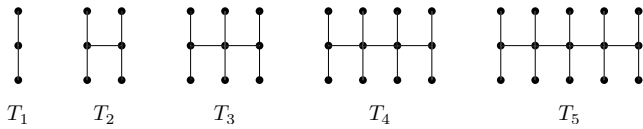
追加問題 2.8 問題 2.1 と同様に、 $a_n$  で頂点数  $n$  の道における独立集合の総数を表すとす。このとき、任意の自然数  $k \geq 1$  に対して、次が成り立つことを証明せよ。

$$a_{2k+1} = a_k^2 + a_{k-1}^2.$$

ヒント：頂点数  $2k+1$  の道を考えて、その「中央」に位置する頂点が独立集合に要素として含まれる場合と含まれない場合を分けて数え上げてみよ。

次ページに続く

追加問題 2.9 自然数  $n \geq 1$  に対して, 次の図で表されるグラフ  $T_n$  を考える.



グラフ  $T_n$  における独立集合の総数を  $t_n$  とするとき, 次の漸化式が成り立つことを証明せよ.

$$t_n = \begin{cases} 5 & (n = 1 \text{ のとき}) \\ 24 & (n = 2 \text{ のとき}) \\ 4t_{n-1} + 4t_{n-2} & (n \geq 3 \text{ のとき}). \end{cases}$$

追加問題 2.10 次のアルゴリズムを考える.

```

1: def fnct2(n)
2:   print "K"
3:   if n == 0
4:     return
5:   elsif n % 2 == 0
6:     fnct2(n/2)
7:   else
8:     fnct2(n-1)
9:   end
10: end

```

自然数  $n \geq 0$  に対して,  $\text{fnct2}(n)$  が出力する  $K$  の総数を  $p_n$  で表す. 次の式が成り立つことを証明せよ.

$$p_n \begin{cases} = 1 & (n = 0 \text{ のとき}) \\ \leq 2 + p_{\lfloor n/2 \rfloor} & (n \geq 1 \text{ のとき}). \end{cases}$$