離散最適化基礎論 (2025 年後学期)

高速指数時間アルゴリズム

第 2 回

分枝アルゴリズム (1)：基礎

岡本 吉央　(電気通信大学)
okamotoy@uec.ac.jp

2025 年 10 月 21 日

最終更新：2025 年 10 月 21 日　12:26

スケジュール (前半)

1. 高速指数時間アルゴリズムの考え方 (10/7)

＊ 休み (体育祭) (10/14)

2. **分枝アルゴリズム：基礎** (10/21)
3. 分枝アルゴリズム：高速化 (10/28)
4. 分枝アルゴリズム：測度統治法 (11/4)
5. 動的計画法：基礎 (11/11)
6. 動的計画法：例 (11/18)

スケジュール (後半)

7.	包除原理：原理	(11/25)
＊	休み (秋ターム試験)	(12/2)
8.	包除原理：例	(12/9)
9.	部分集合たたみ込み：原理	(12/16)
＊	休み (出張)	(12/23)
＊	休み (冬季休業)	(12/30)
10.	部分集合たたみ込み：例	(1/6)
11.	指数時間仮説：原理	(1/13)
12.	指数時間仮説：証明	(1/20)
13.	最近の話題	(1/27)
＊	休み (修士論文発表会)	(2/3)

本日の内容

1. **分枝アルゴリズムの設計：基礎**
2. 分枝アルゴリズムの解析：基礎
3. 他の例：充足可能性問題

分枝アルゴリズム (branching algorithm) とは？

場合分け に基づくアルゴリズム
場合分けを根つき木 (探索木) で表すことが多い

他の呼び名

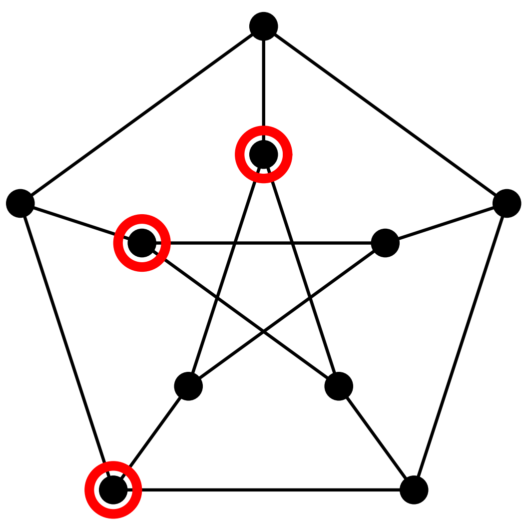
- 分枝限定法
- 探索木アルゴリズム
- Davis-Putnam 型アルゴリズム
- ...

(復習) 独立集合

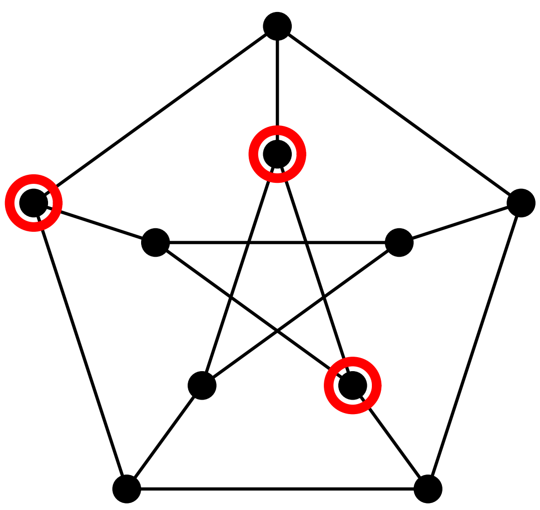
無向グラフ $G = (V, E)$

定義：独立集合 (independent set)

G の **独立集合** とは，頂点部分集合 $S \subseteq V$ で，
どの 2 頂点 $u, v \in S$ も隣接していないもの



独立集合である



独立集合ではない

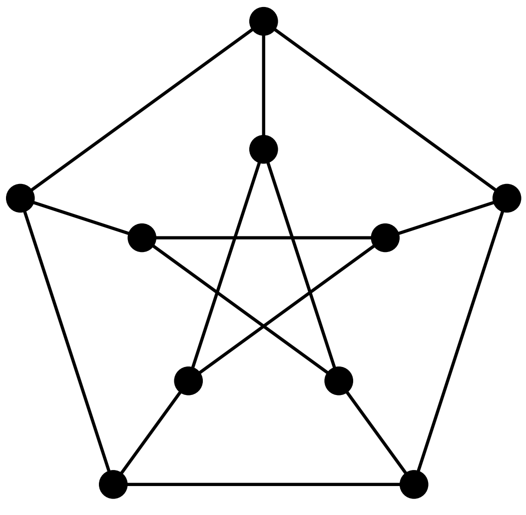
(復習) 最大独立集合問題

問題：最大独立集合問題

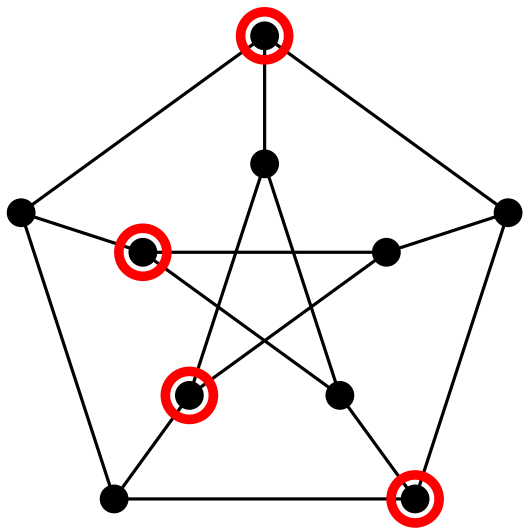
入力： 無向グラフ $G = (V, E)$

出力： G の最大独立集合

要素数最大の独立集合



入力

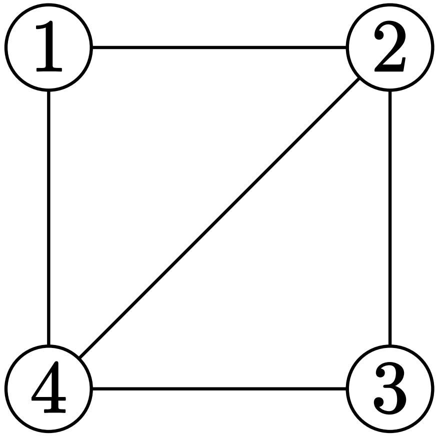


出力

注：最大独立集合問題は NP 困難 (Karp '72)

(復習) 頂点の次数

定義：頂点 v の **次数** (degree) とは，v に隣接する頂点の数
$\deg(v)$ で表すことがある



- $\deg(1) = 2$
- $\deg(2) = 3$
- $\deg(3) = 2$
- $\deg(4) = 3$

このグラフにおける
- 最大次数 $= 3$
- 最小次数 $= 2$

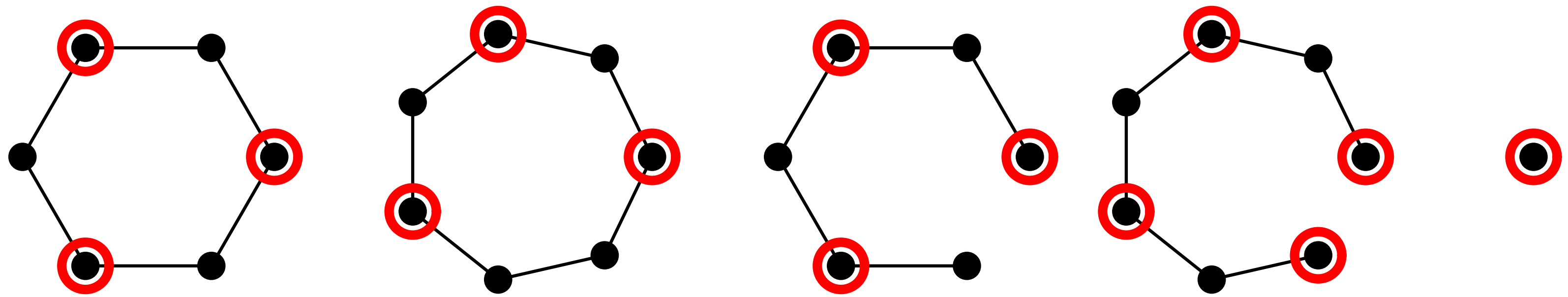
定義：頂点 v の **開近傍** $N(v) =$ 頂点 v の隣接頂点全体の集合
頂点 v の **閉近傍** $N[v] = N(v) \cup \{v\}$

- $N(3) = \{2, 4\}$，$N[3] = \{2, 3, 4\}$

(復習) 工夫：補題

補題 1

最大次数が 2 以下のグラフに対して，
最大独立集合問題は線形時間で解ける

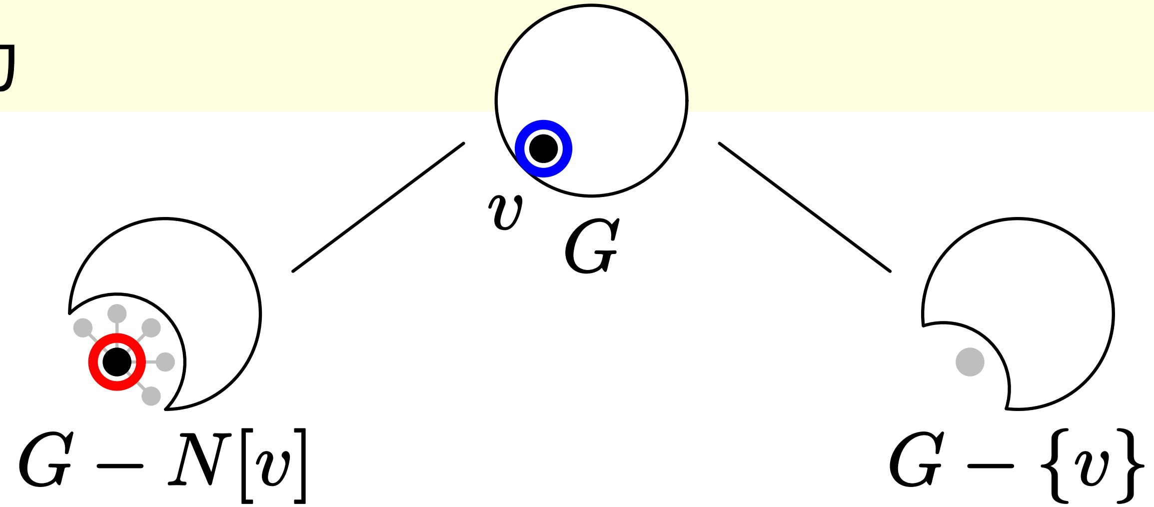


(復習) 工夫したアルゴリズム

アルゴリズム A(G)

1. if G の最大次数 ≤ 2:
   - 補題 1 を使って，最大独立集合を出力
2. $v = G$ の次数最大の頂点
3. 次の 2 つの大きいほうを出力
   - A($G - N[v]$) の出力 $\cup \{v\}$
   - A($G - \{v\}$) の出力

$G - X = G$ から X を除去してできるグラフ



ステップ 3 の正当性 (1)

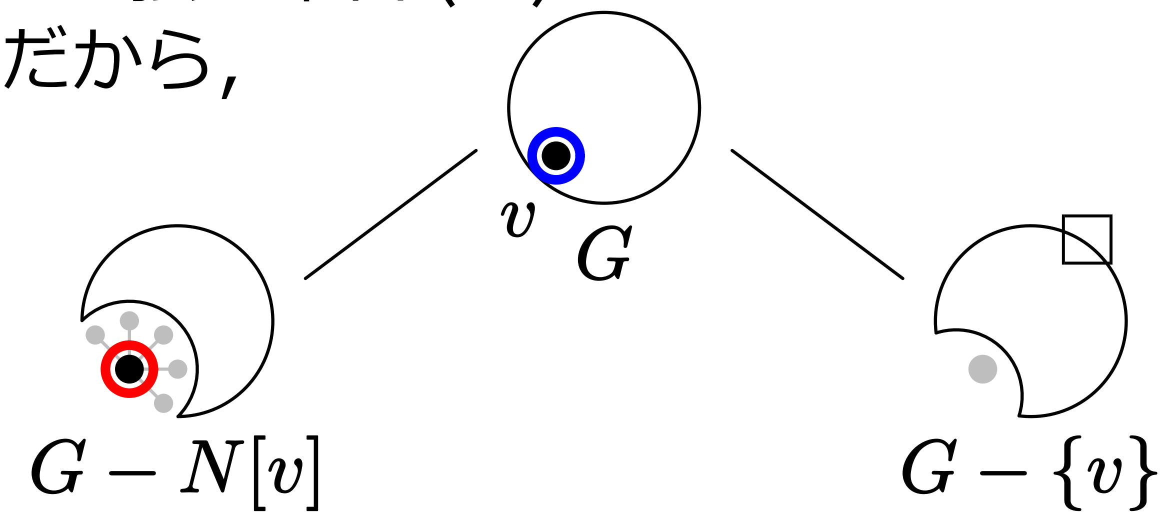
補題 2

無向グラフ $G = (V, E)$ の
任意の最大独立集合 S と任意の頂点 v に対して

1. $v \in S \Rightarrow S - \{v\}$ は $G - N[v]$ の最大独立集合
2. $v \notin S \Rightarrow S$ は $G - \{v\}$ の最大独立集合

<u>証明 (1)</u> : $v \in S$ のとき，$S - \{v\}$ は $G - N[v]$ の独立集合 (?)

- T を $G - N[v]$ の任意の最大独立集合とする
- このとき，$T \cup \{v\}$ は G の独立集合 (?)
- S が G の最大独立集合だから，
  $|S| \geq |T \cup \{v\}|$
- $\therefore |S - \{v\}| \geq |T|$



ステップ 3 の正当性 (2)

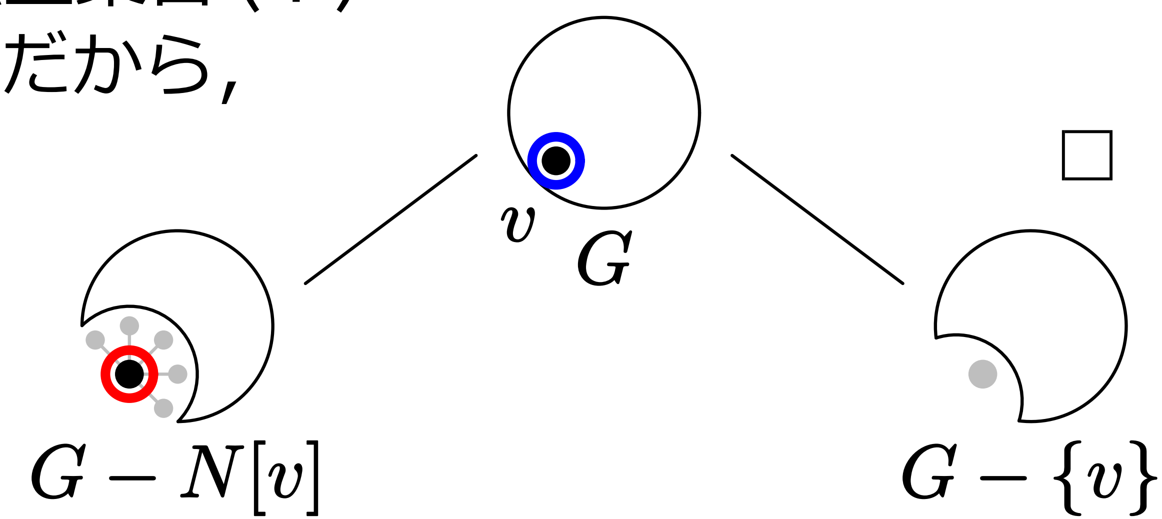
補題 2

無向グラフ $G = (V, E)$ の
任意の最大独立集合 S と任意の頂点 v に対して

1. $v \in S \Rightarrow S - \{v\}$ は $G - N[v]$ の最大独立集合
2. $v \notin S \Rightarrow S$ は $G - \{v\}$ の最大独立集合

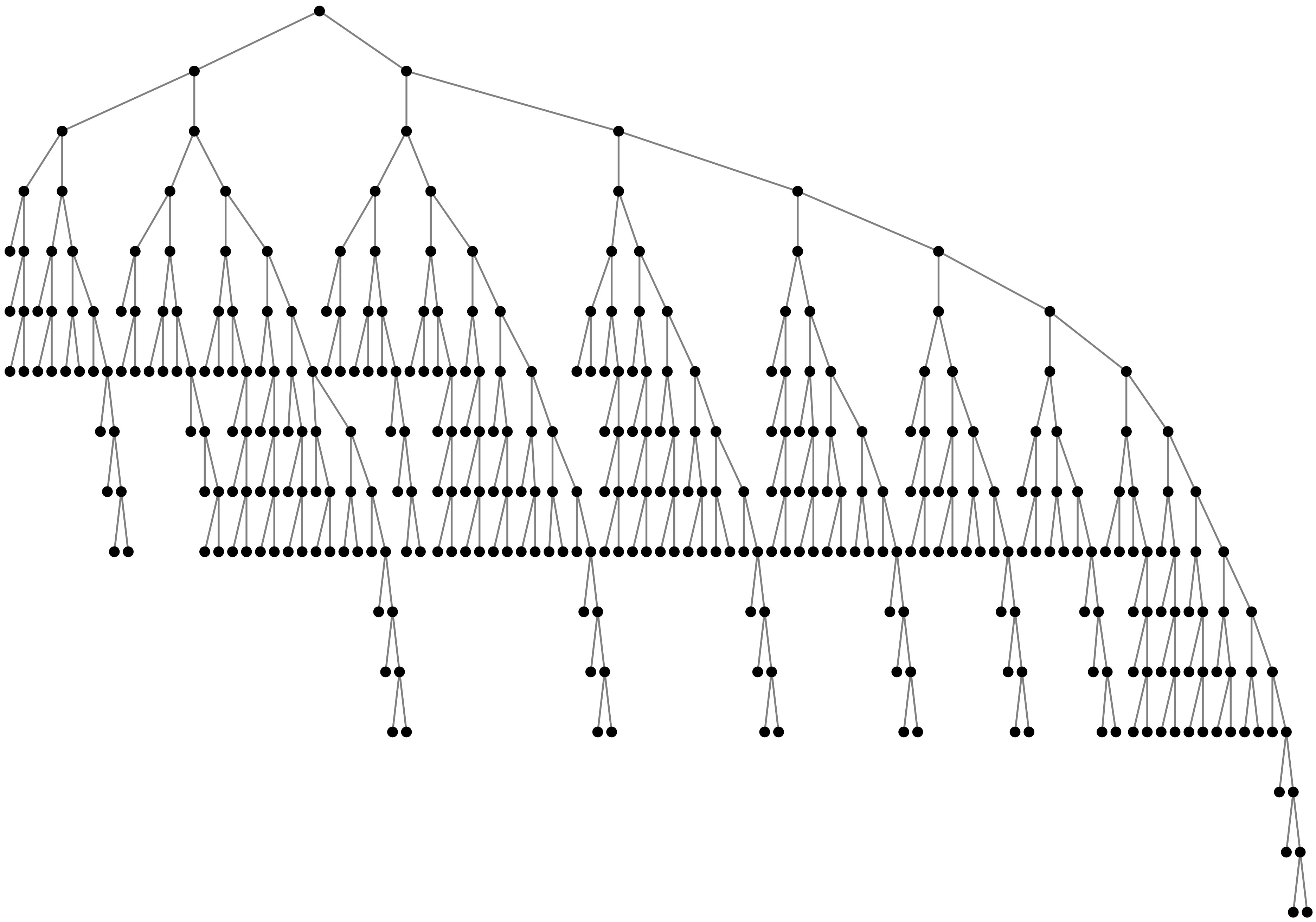
証明 (2)：$v \notin S$ のとき，S は $G - \{v\}$ の独立集合 (?)

- T を $G - \{v\}$ の任意の最大独立集合とする
- このとき，T は G の独立集合 (?)
- S は G の最大独立集合だから，
  $|S| \geq |T|$

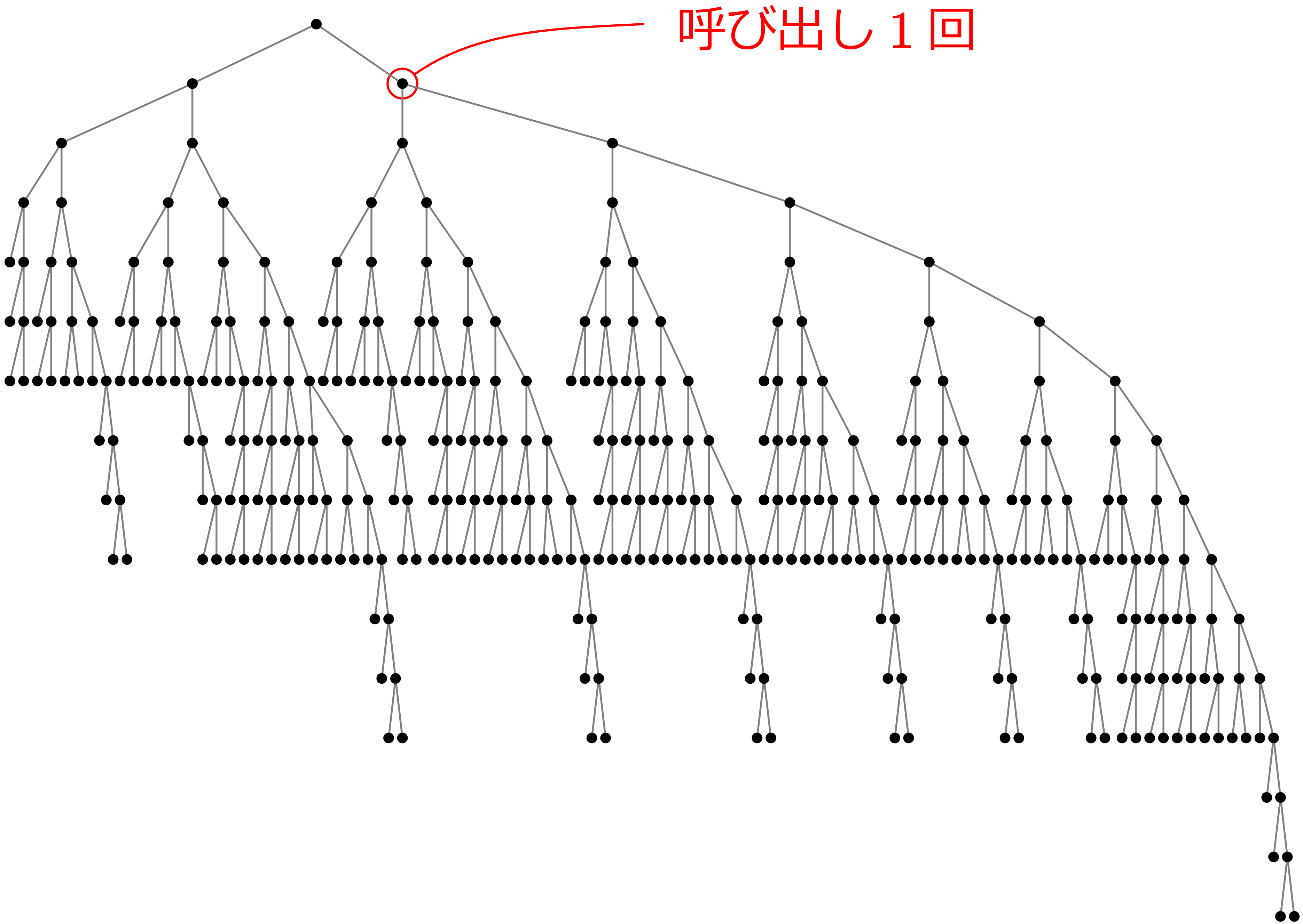


□

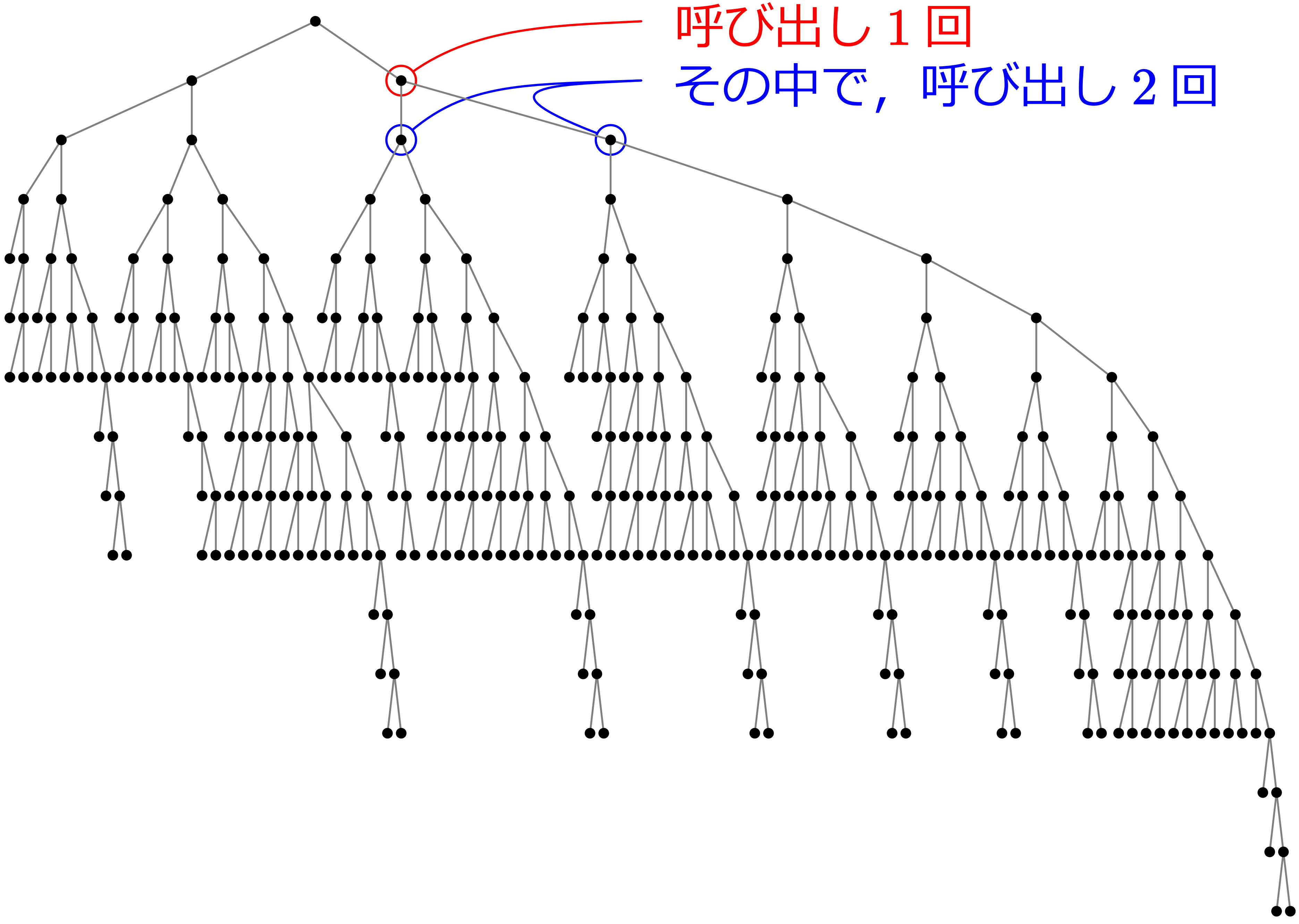
探索木 (search tree)



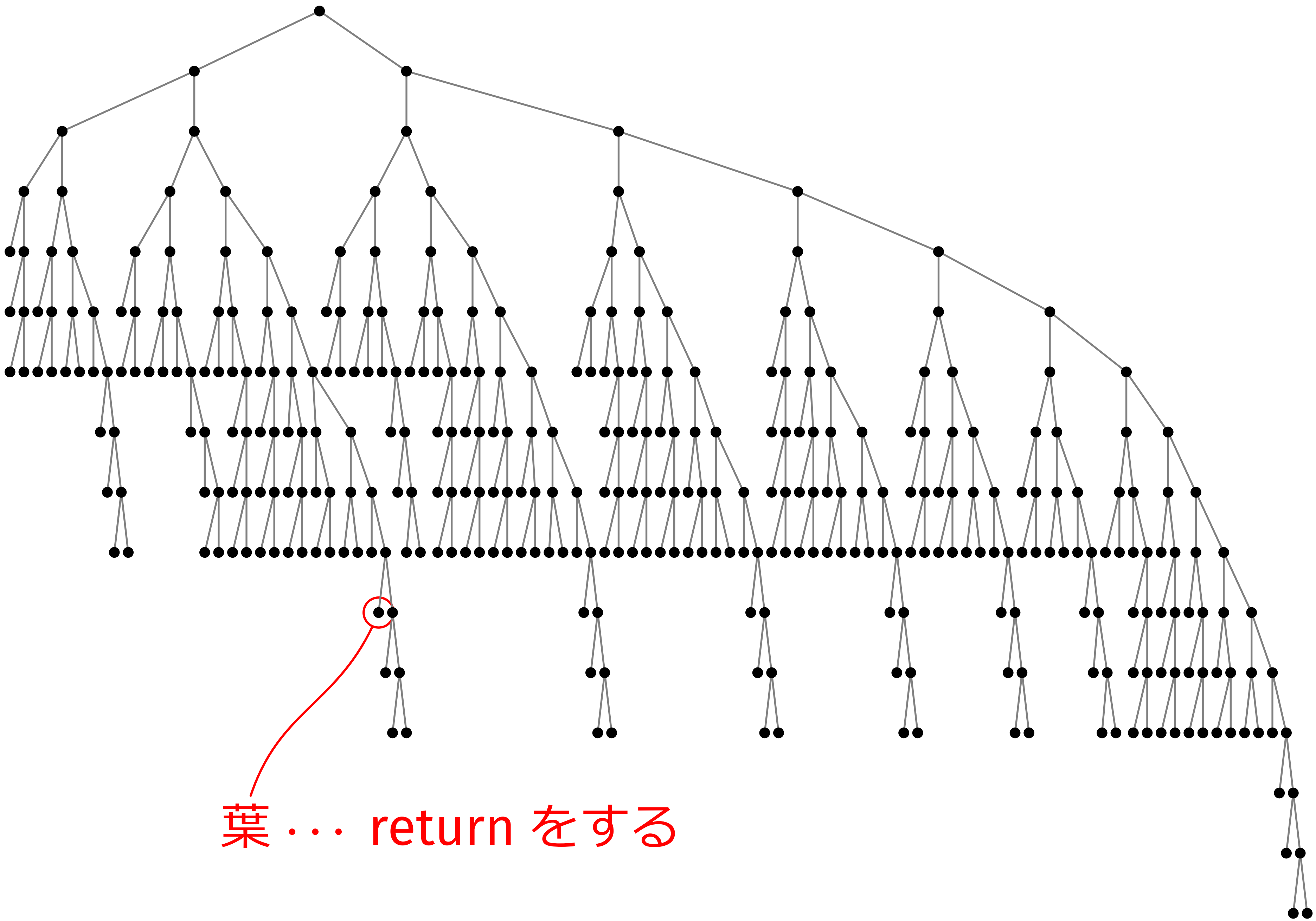
探索木 (search tree)



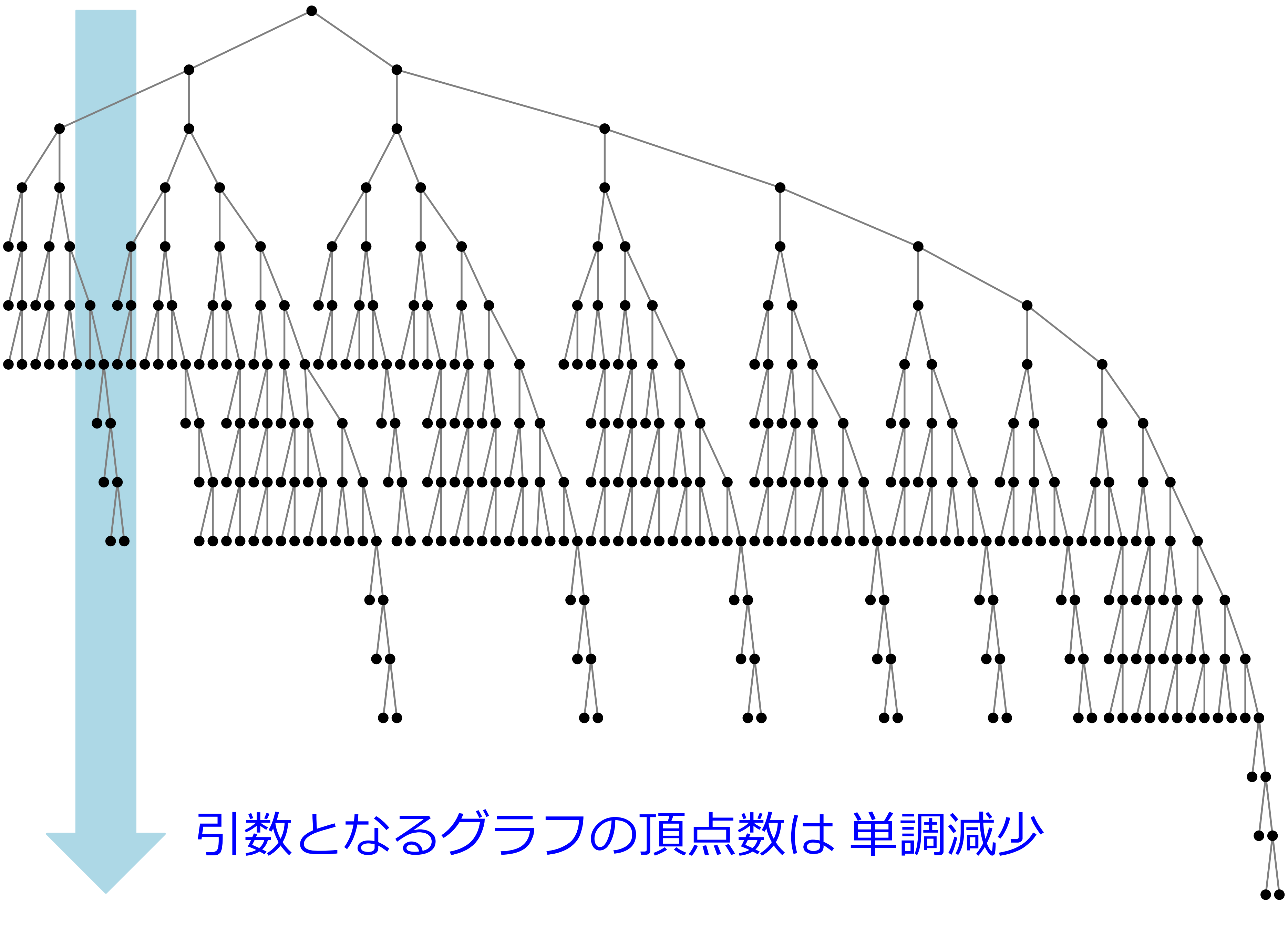
探索木 (search tree)



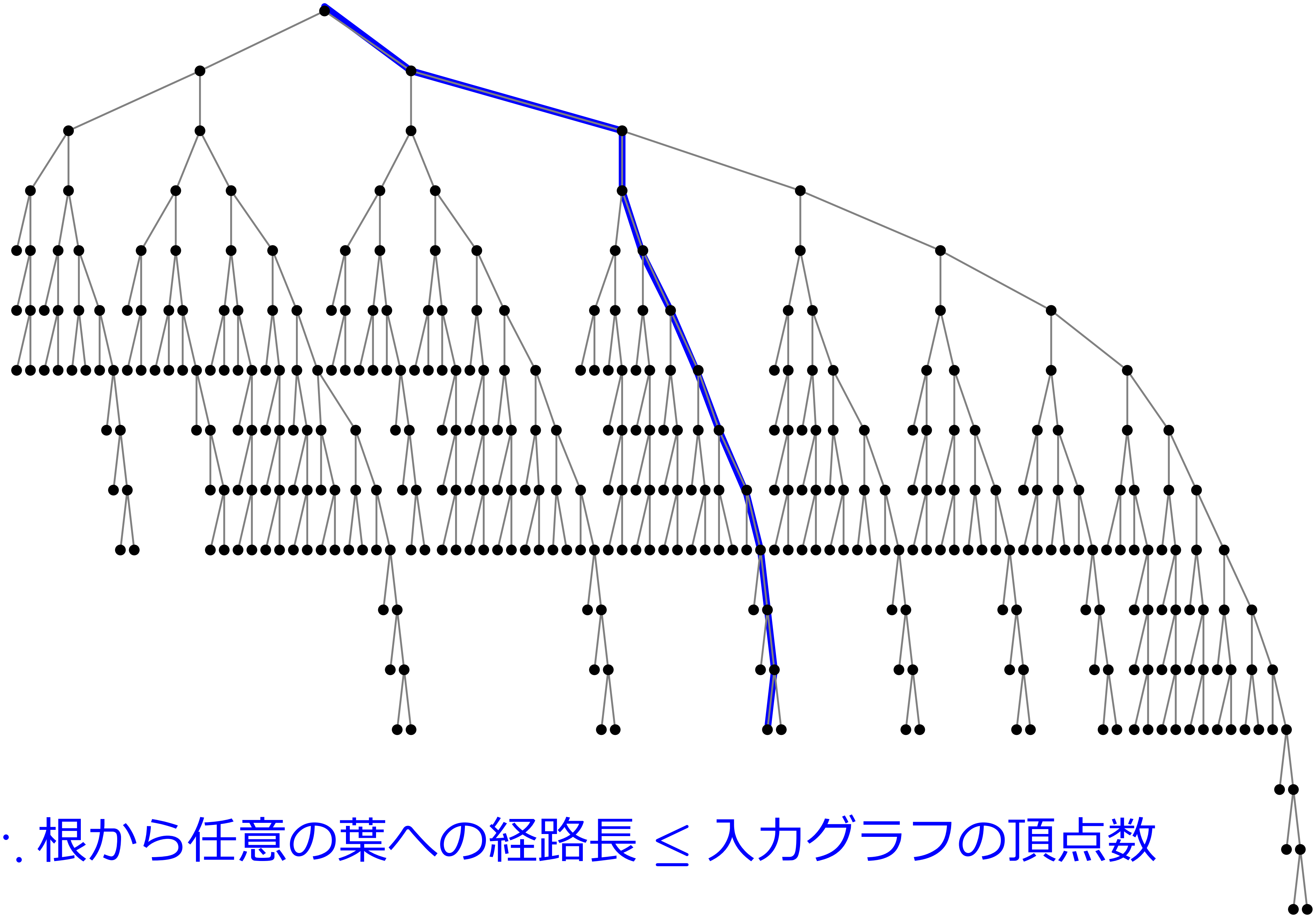
探索木 (search tree)



探索木 (search tree)

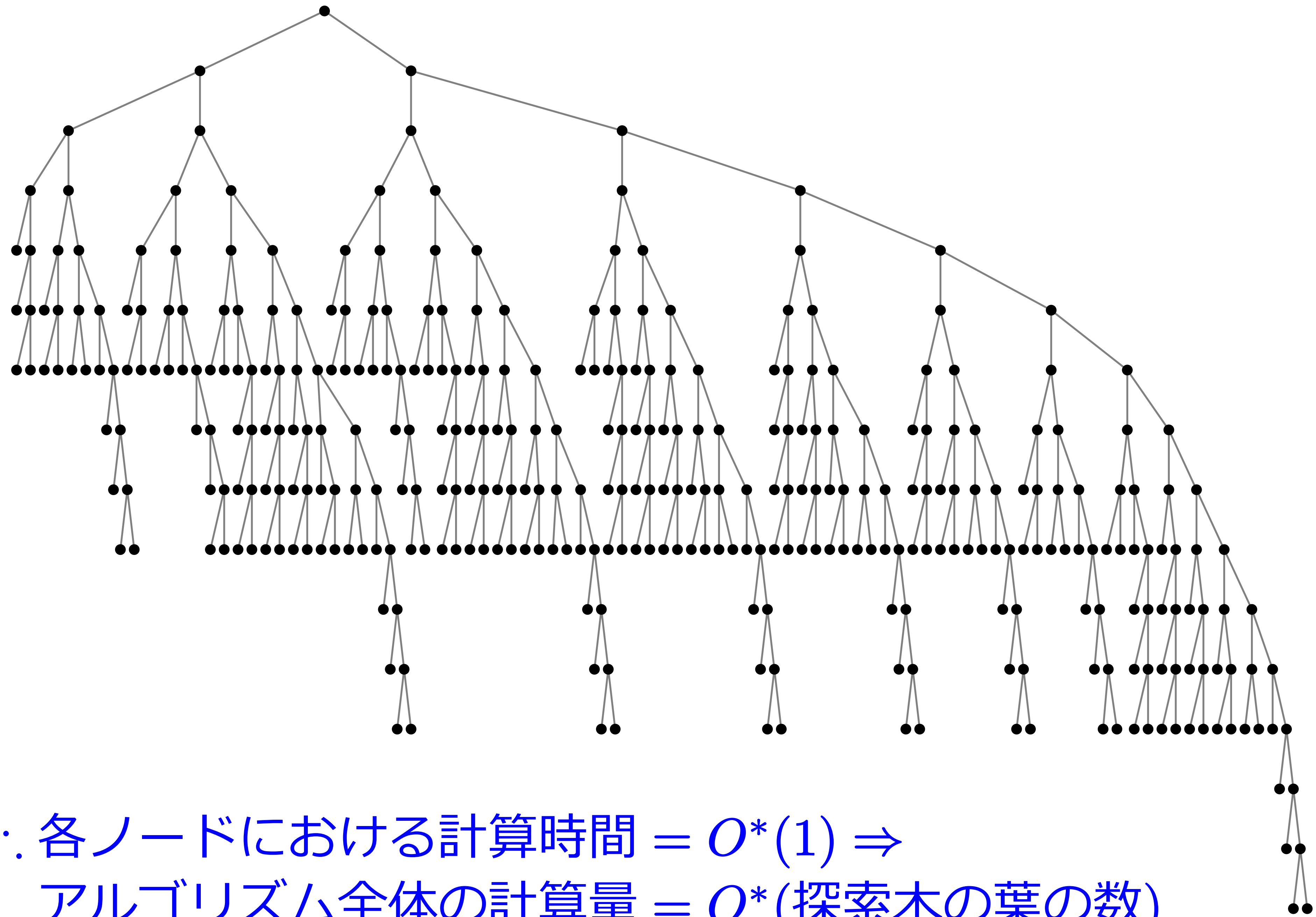


探索木 (search tree)



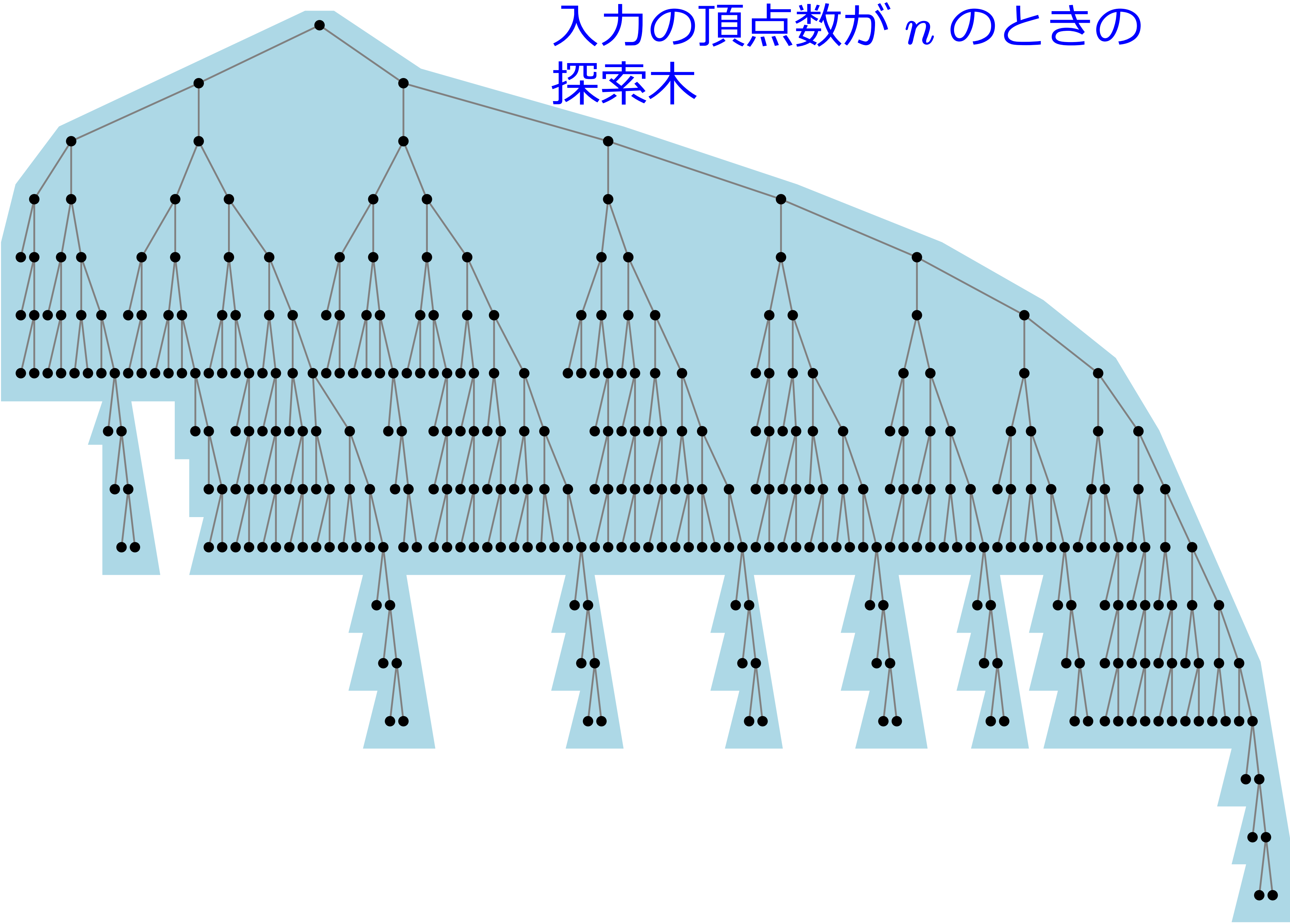
∴ 根から任意の葉への経路長 ≤ 入力グラフの頂点数

探索木 (search tree)



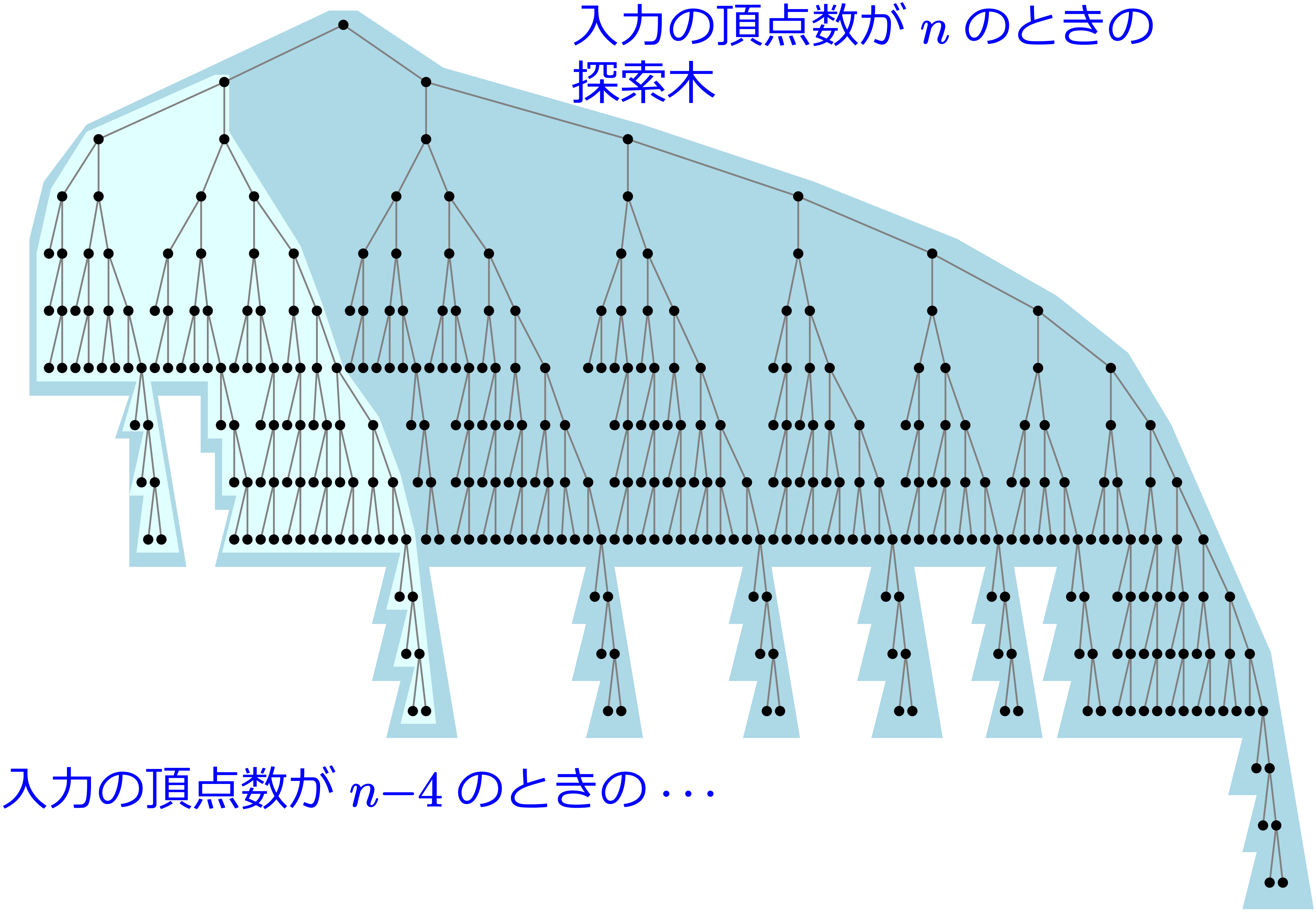
$\therefore$ 各ノードにおける計算時間 $= O^*(1) \Rightarrow$
アルゴリズム全体の計算量 $= O^*($探索木の葉の数$)$

探索木 (search tree)

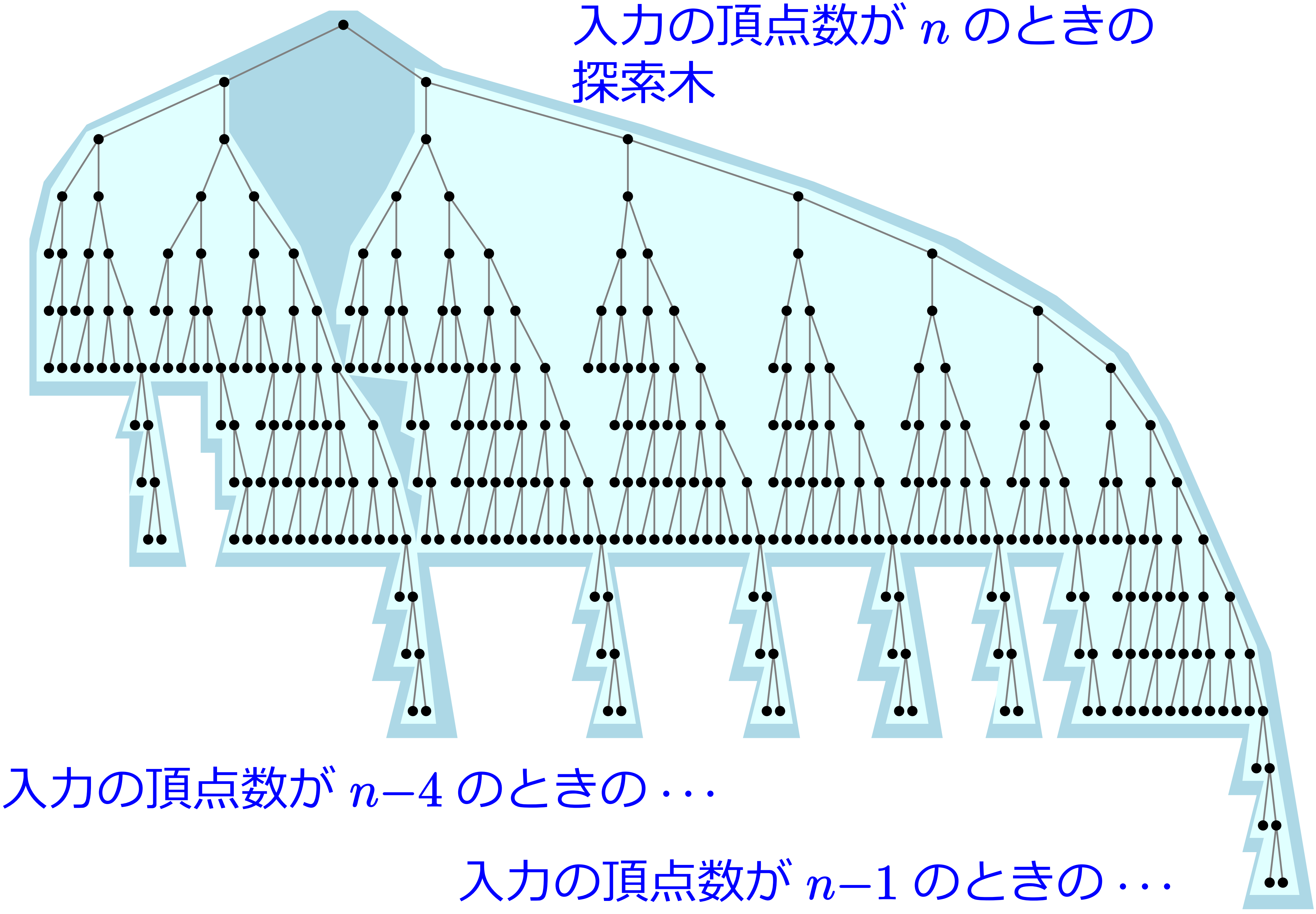


入力の頂点数が n のときの探索木

探索木 (search tree)



探索木 (search tree)

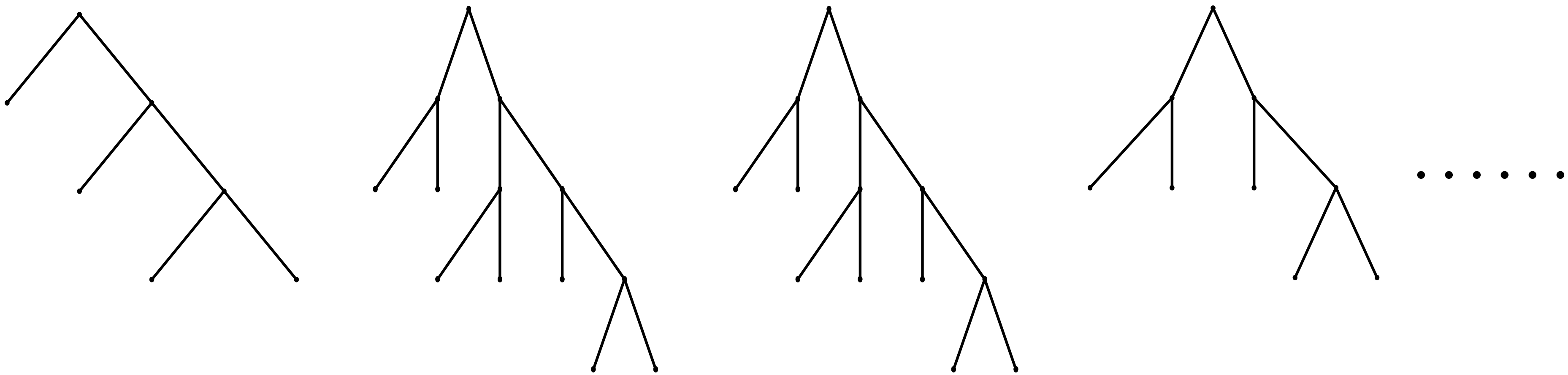


探索木の葉の数

記法

アルゴリズム A(·) に対して，次を定義

- $T(G)$ = 入力を G としたときの探索木の葉の数
- $T(n) = \max\{T(G) \mid G$ の頂点数 $\leq n\}$



$n = 10$

記法

アルゴリズム A(·) に対して，次を定義

- $T(G)=$ 入力を G としたときの探索木の葉の数
- $T(n)=\max\{T(G) \mid G$ の頂点数 $\leq n\}$

ほしいもの：$T(n)$ に対する小さな上界

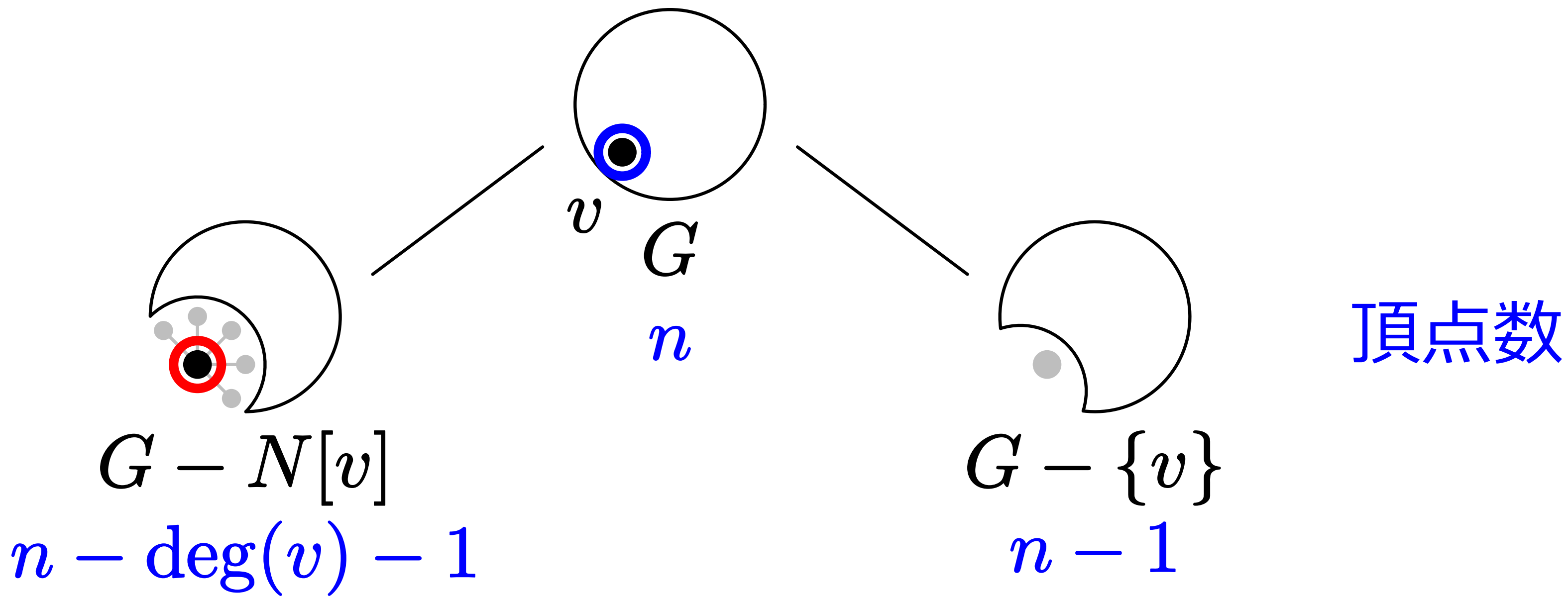
性質

$n \leq n' \Rightarrow T(n) \leq T(n')$

正しいことは，定義から直ちに分かる

探索木の葉の数：漸化式

$n \geq 4$ として，$T(n) = T(G)$ を満たすグラフ G を考える



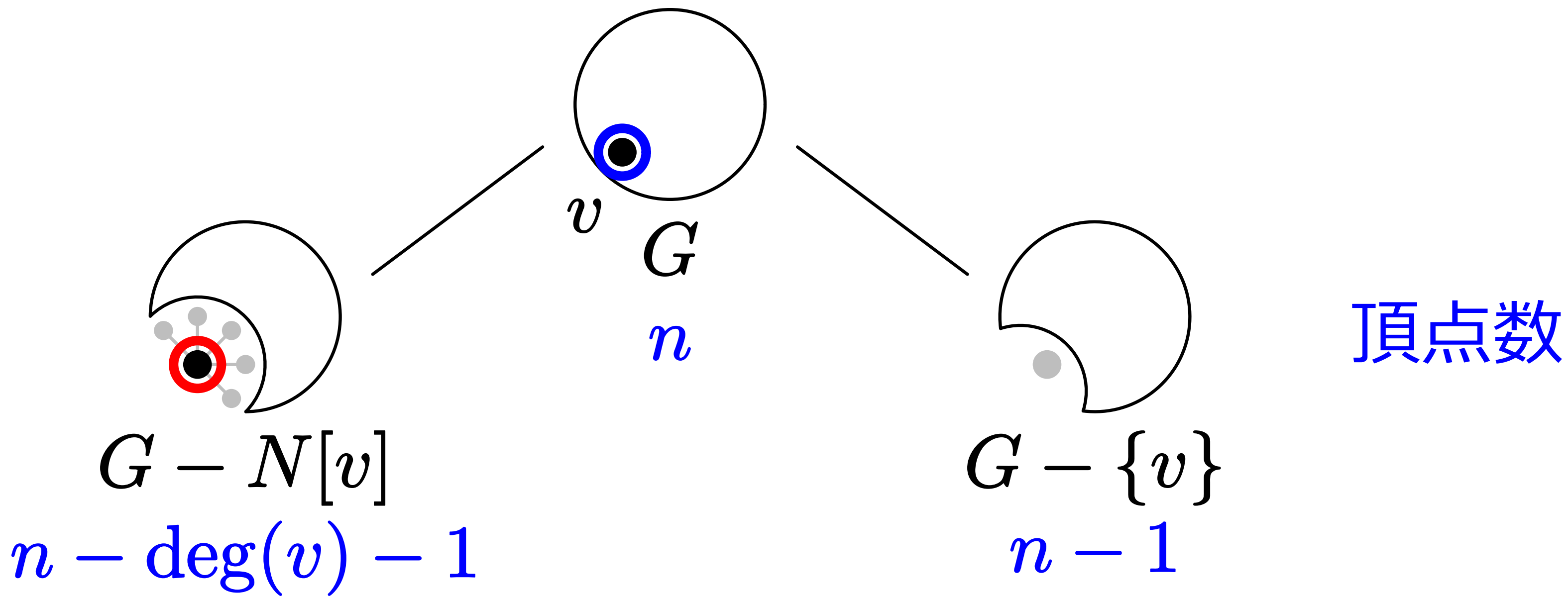
記法

アルゴリズム A(·) に対して，次を定義

- $T(G)$ = 入力を G としたときの探索木の葉の数
- $T(n) = \max\{T(G) \mid G$ の頂点数 $\leq n\}$

探索木の葉の数：漸化式

$n \geq 4$ として，$T(n) = T(G)$ を満たすグラフ G を考える



$$
\begin{aligned}
T(n) = T(G) &= T(G - N[v]) + T(G - \{v\}) \\
&\leq T(n - \deg(v) - 1) + T(n - 1) \\
&\leq T(n - 4) + T(n - 1)
\end{aligned}
$$

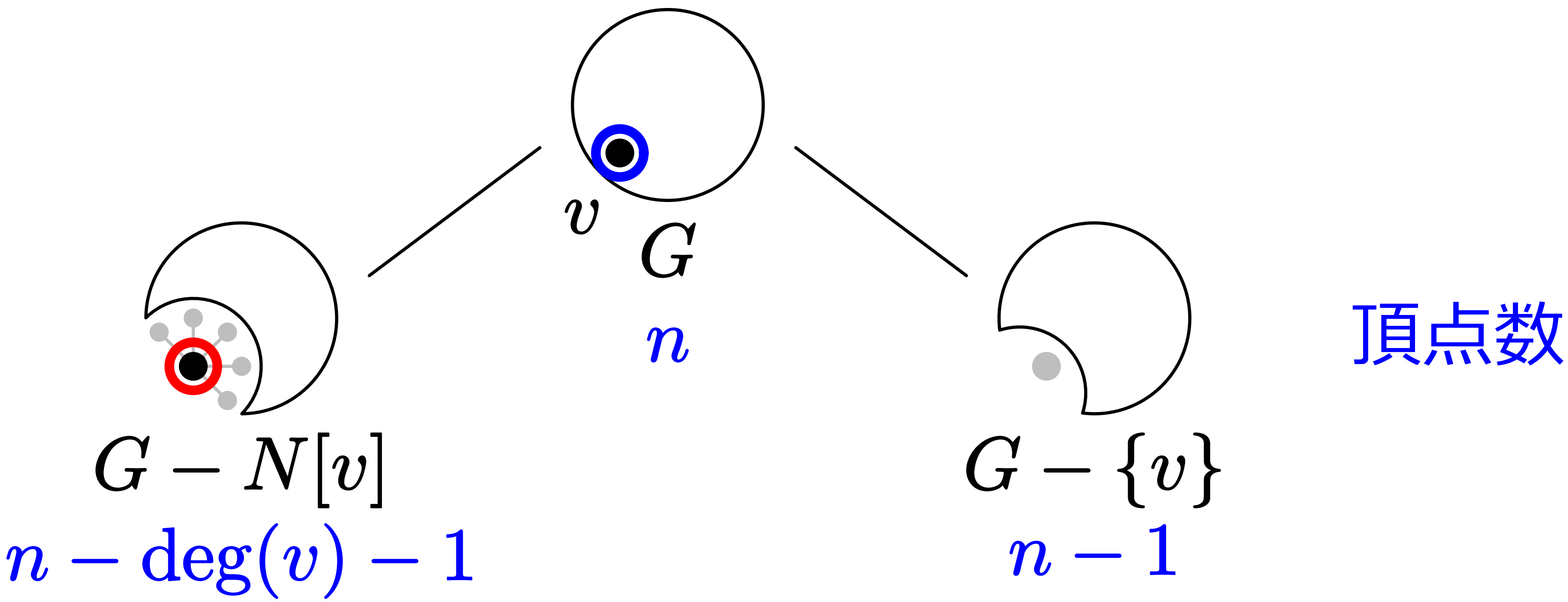
記法

アルゴリズム A(·) に対して，次を定義

- $T(G)$ = 入力を G としたときの探索木の葉の数
- $T(n) = \max\{T(G) \mid G$ の頂点数 $\leq n\}$

探索木の葉の数：漸化式

$n \geq 4$ として，$T(n) = T(G)$ を満たすグラフ G を考える

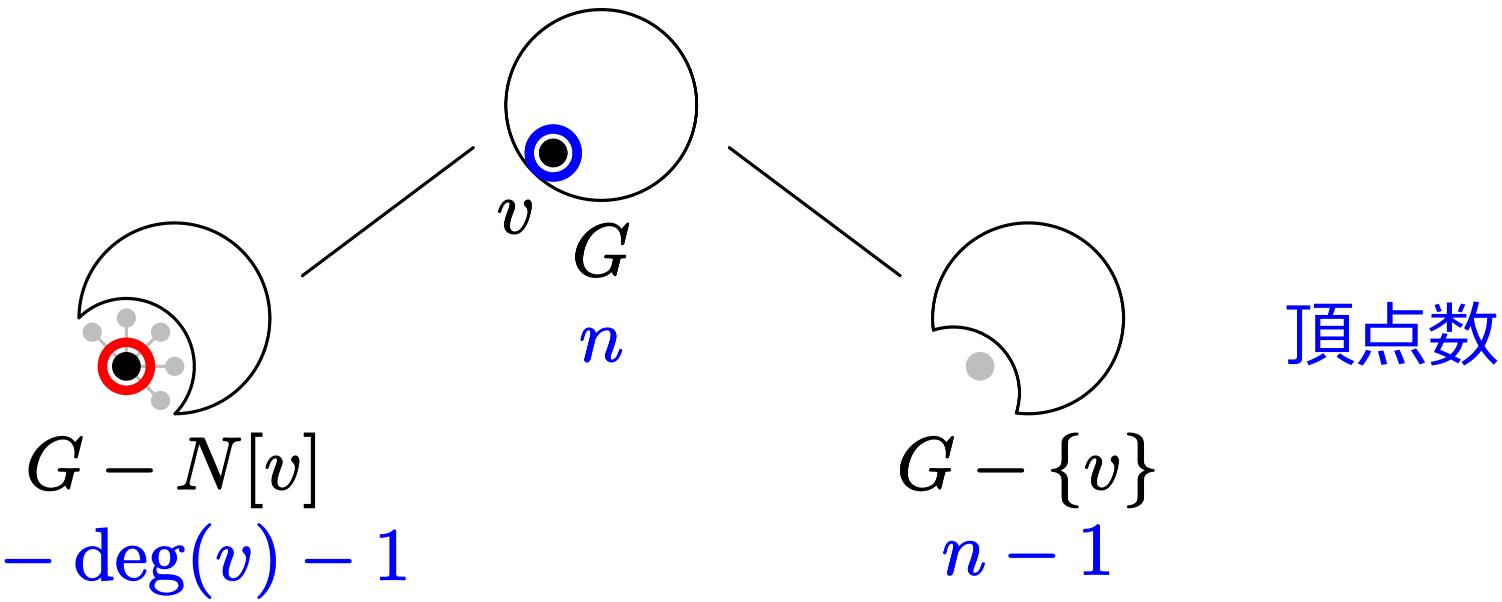


$$
\begin{aligned}
T(n) = T(G) &= T(G - N[v]) + T(G - \{v\}) \\
&\leq T(n - \deg(v) - 1) + T(n - 1) \\
&\leq T(n - 4) + T(n - 1)
\end{aligned}
$$

$\therefore T(n) = \underline{O(1.3803^n)}$ (導出は次節)

探索木の葉の数：漸化式

$n \geq 4$ として，$T(n) = T(G)$ を満たすグラフ G を考える



$$
\begin{aligned}
T(n) = T(G) &= T(G - N[v]) + T(G - \{v\}) \\
&\leq T(n - \deg(v) - 1) + T(n - 1) \\
&\leq T(n - 4) + T(n - 1)
\end{aligned}
$$

$\therefore T(n) = O(1.3803^n)$　（導出は次節）

$\boxed{\text{結論}}$：アルゴリズム A(G) の計算量は $O^*(1.3803^n)$

本日の内容

1. 分枝アルゴリズムの設計：基礎
2. **分枝アルゴリズムの解析：基礎**
3. 他の例：充足可能性問題

漸化式から上界を導出する

目標

次を満たす数列 $T(0), T(1), \ldots, T(n), \ldots$ に対して，
$T(n)$ のよい上界を導出すること

$$T(n) \leq \begin{cases} T(n-4) + T(n-1) & (n \geq 4 \text{ のとき}) \\ c & (n \leq 3 \text{ のとき}) \end{cases}$$

ただし，c はある非負定数

漸化式から上界を導出する

目標

次を満たす数列 $T(0), T(1), \ldots, T(n), \ldots$ に対して，
$T(n)$ のよい上界を導出すること

$$T(n) \leq \begin{cases} T(n-4) + T(n-1) & (n \geq 4 \text{ のとき}) \\ c & (n \leq 3 \text{ のとき}) \end{cases}$$

ただし，c はある非負定数

手順

1. 「$\leq$」を「$=$」に変える
2. 特性方程式を解く

上界の導出：手順 1

$$T(n) \le \begin{cases} T(n-4) + T(n-1) & (n \ge 4 \text{ のとき}) \\ c & (n \le 3 \text{ のとき}) \end{cases}$$

$$\tilde{T}(n) = \begin{cases} \tilde{T}(n-4) + \tilde{T}(n-1) & (n \ge 4 \text{ のとき}) \\ c & (n \le 3 \text{ のとき}) \end{cases}$$

上界の導出：手順 1

$$T(n) \le \begin{cases} T(n-4) + T(n-1) & (n \ge 4 \text{ のとき}) \\ c & (n \le 3 \text{ のとき}) \end{cases}$$

$$\downarrow$$

$$\tilde{T}(n) = \begin{cases} \tilde{T}(n-4) + \tilde{T}(n-1) & (n \ge 4 \text{ のとき}) \\ c & (n \le 3 \text{ のとき}) \end{cases}$$

性質

任意の整数 $n \ge 0$ に対して，$T(n) \le \tilde{T}(n)$

証明：n に関する数学的帰納法 $\square$

上界の導出：手順1

$$T(n) \le \begin{cases} T(n-4) + T(n-1) & (n \ge 4 \text{ のとき}) \\ c & (n \le 3 \text{ のとき}) \end{cases}$$

$$\downarrow$$

$$\tilde{T}(n) = \begin{cases} \tilde{T}(n-4) + \tilde{T}(n-1) & (n \ge 4 \text{ のとき}) \\ c & (n \le 3 \text{ のとき}) \end{cases}$$

性質

任意の整数 $n \ge 0$ に対して，$T(n) \le \tilde{T}(n)$

証明：n に関する数学的帰納法 □

帰結：$\tilde{T}(n)$ のよい上界を導出すれば十分

上界の導出 (補足) : $\tilde{T}(n)$ を調べる

$$\tilde{T}(n) = \begin{cases} \tilde{T}(n-4) + \tilde{T}(n-1) & (n \geq 4 \text{ のとき}) \\ c & (n \leq 3 \text{ のとき}) \end{cases}$$

n	$\tilde{T}(n)$
0	c
1	c
2	c
3	c
4	$2c$
5	$3c$
6	$4c$
7	$5c$
8	$7c$
9	$10c$

n	$\tilde{T}(n)$
10	$14c$
11	$19c$
12	$26c$
13	$36c$
14	$50c$
15	$69c$
16	$95c$
17	$131c$
18	$181c$
19	$250c$

上界の導出：手順 2

$$\tilde{T}(n) = \begin{cases} \tilde{T}(n-4) + \tilde{T}(n-1) & (n \geq 4 \text{ のとき}) \\ c & (n \leq 3 \text{ のとき}) \end{cases}$$

$n \geq 4$ のとき, $\boxed{\tilde{T}(n) = ax^n}$ とおく　$(a > 0, x > 0)$

このとき,

$$ax^n = ax^{n-4} + ax^{n-1}$$

上界の導出：手順 2

$$\tilde{T}(n) = \begin{cases} \tilde{T}(n-4) + \tilde{T}(n-1) & (n \geq 4 \text{ のとき}) \\ c & (n \leq 3 \text{ のとき}) \end{cases}$$

$n \geq 4$ のとき，$\boxed{\tilde{T}(n) = ax^n}$ とおく　$(a > 0, x > 0)$

このとき，

$$ax^n = ax^{n-4} + ax^{n-1}$$

$/ax^{n-4}$

$$x^4 = 1 + x^3$$

上界の導出：手順 2

$$\tilde{T}(n) = \begin{cases} \tilde{T}(n-4) + \tilde{T}(n-1) & (n \geq 4 \text{ のとき}) \\ c & (n \leq 3 \text{ のとき}) \end{cases}$$

$n \geq 4$ のとき, $\boxed{\tilde{T}(n) = ax^n}$ とおく　$(a > 0, x > 0)$

このとき,

$$ax^n = ax^{n-4} + ax^{n-1}$$

$$\boxed{x^4 = 1 + x^3}$$

$/ax^{n-4}$

この方程式を (数値的に) 解くと,

$$x = 1.3803, -0.8191, 0.2195 \pm 0.9145i \quad (i \text{ は虚数単位})$$

$$\tilde{T}(n) = \begin{cases} \tilde{T}(n-4) + \tilde{T}(n-1) & (n \geq 4 \text{ のとき}) \\ c & (n \leq 3 \text{ のとき}) \end{cases}$$

$n \geq 4$ のとき, $\boxed{\tilde{T}(n) = ax^n}$ とおく $\quad (a > 0, x > 0)$

このとき,

$$ax^n = ax^{n-4} + ax^{n-1}$$

$/ax^{n-4}$

$$\boxed{x^4 = 1 + x^3}$$

この方程式を (数値的に) 解くと,

$$x = 1.3803, -0.8191, 0.2195 \pm 0.9145i \quad (i \text{ は虚数単位})$$

$\boxed{\text{結論}}$: $\tilde{T}(n) = O(1.3803^n)$ $\quad \leftarrow$ なぜ？

上界の導出：正しい理由

$$\tilde{T}(n) = \begin{cases} \tilde{T}(n-4) + \tilde{T}(n-1) & (n \geq 4 \text{ のとき}) \\ c & (n \leq 3 \text{ のとき}) \end{cases}$$

証明したいこと

λ を $x^4 = 1 + x^3$ の正実数解とする
このとき，$\tilde{T}(n) = O(\lambda^n)$ である

上界の導出：正しい理由

$$\tilde{T}(n) = \begin{cases} \tilde{T}(n-4) + \tilde{T}(n-1) & (n \geq 4 \text{ のとき}) \\ c & (n \leq 3 \text{ のとき}) \end{cases}$$

証明したいこと

λ を $x^4 = 1 + x^3$ の正実数解とする
このとき，$\tilde{T}(n) = O(\lambda^n)$ である

O 記法の定義を使って書き直し

ある整数 $n_0 \geq 0$ と実数 $a > 0$ が存在して，
任意の整数 $n \geq n_0$ に対して，$\tilde{T}(n) \leq a \cdot \lambda^n$.

$$\tilde{T}(n) = \begin{cases} \tilde{T}(n-4) + \tilde{T}(n-1) & (n \geq 4 \text{ のとき}) \\ c & (n \leq 3 \text{ のとき}) \end{cases}$$

証明したいこと

λ を $x^4 = 1 + x^3$ の正実数解とする
このとき，$\tilde{T}(n) = O(\lambda^n)$ である

O 記法の定義を使って書き直し

ある整数 $n_0 \geq 0$ と実数 $a > 0$ が存在して，
任意の整数 $n \geq n_0$ に対して，$\tilde{T}(n) \leq a \cdot \lambda^n$.

証明：$n_0 = 0$，$a = c$ とする

上界の導出：正しい理由 (続)

証明 (続)：$n \geq n_0 = 0$ に関する帰納法

$n = 0, 1, 2, 3$ のとき

- $\tilde{T}(n) = c = a \leq a \cdot \lambda^n$

$\lambda > 1$

上界の導出：正しい理由 (続)

<u>証明 (続)</u>：$n \geq n_0 = 0$ に関する帰納法

$n = 0, 1, 2, 3$ のとき

- $\tilde{T}(n) = c = a \leq a \cdot \lambda^n$

任意の整数 $k \geq 3$ を考える

- $0 \leq \ell \leq k$ を満たす任意の整数 ℓ に対して $\tilde{T}(\ell) \leq a \cdot \lambda^\ell$ が成り立つと仮定

上界の導出：正しい理由 (続)

<u>証明 (続)</u>：$n \ge n_0 = 0$ に関する帰納法

$n = 0, 1, 2, 3$ のとき

- $\tilde{T}(n) = c = a \le a \cdot \lambda^n$

任意の整数 $k \ge 3$ を考える

- $0 \le \ell \le k$ を満たす任意の整数 ℓ に対して $\tilde{T}(\ell) \le a \cdot \lambda^\ell$ が成り立つと仮定
- このとき，$k+1 \ge 4$ なので，
$$\begin{aligned}\tilde{T}(k+1) &= \tilde{T}(k-3) + \tilde{T}(k) \le a \cdot \lambda^{k-3} + a \cdot \lambda^k \\ &= a \cdot \lambda^{k-3}(1+\lambda^3) = a \cdot \lambda^{k-3} \cdot \lambda^4 = a \cdot \lambda^{k+1}\end{aligned}$$

帰納法の仮定

$\lambda^4 = 1 + \lambda^3$

証明 (続)：$n \geq n_0 = 0$ に関する帰納法

$n = 0, 1, 2, 3$ のとき

- $\tilde{T}(n) = c = a \leq a \cdot \lambda^n$

任意の整数 $k \geq 3$ を考える

- $0 \leq \ell \leq k$ を満たす任意の整数 ℓ に対して $\tilde{T}(\ell) \leq a \cdot \lambda^\ell$ が成り立つと仮定
- このとき，$k + 1 \geq 4$ なので，
$$\begin{aligned}\tilde{T}(k+1) &= \tilde{T}(k-3) + \tilde{T}(k) \leq a \cdot \lambda^{k-3} + a \cdot \lambda^k \\ &= a \cdot \lambda^{k-3}(1 + \lambda^3) = a \cdot \lambda^{k-3} \cdot \lambda^4 = a \cdot \lambda^{k+1}\end{aligned}$$

したがって，任意の整数 $n \geq n_0$ に対して，$\tilde{T}(n) \leq a \cdot \lambda^n$ $\square$

上界の導出：まとめ

$$T(n) \le \begin{cases} T(n-4) + T(n-1) & (n \ge 4 \text{ のとき}) \\ c & (n \le 3 \text{ のとき}) \end{cases}$$

上界を導出するだけなら，次を行えば十分

特性方程式 $x^4 = 1 + x^3$ の正実数解が $\lambda \Rightarrow$

$$T(n) = O(\lambda^n)$$

本日の内容

1. 分枝アルゴリズムの設計：基礎
2. 分枝アルゴリズムの解析：基礎
3. **他の例：充足可能性問題**

問題：充足可能性問題

入力： 論理式 φ

出力： φ を 1 (真) とする割当がある $\Rightarrow$ Yes
φ を 1 (真) とする割当がない $\Rightarrow$ No

充足可能性問題：satisfiability problem (SAT)（サット）

扱う論理式の種類を制限する場合が多い (後述)

論理式 (命題論理式)

$$\varphi = (x_1 \wedge \underbrace{\overline{x_2}}_{}) \vee (x_1 \vee \overline{x_3} \vee x_4)$$

NOT

AND

OR

x	y	$x \wedge y$	$x \vee y$
0	0	0	0
0	1	0	1
1	0	0	1
1	1	1	1

x	$\overline{x}$
0	1
1	0

論理式 (命題論理式)

NOT

$$\varphi = (x_1 \wedge \overline{x_2}) \vee (x_1 \vee \overline{x_3} \vee x_4)$$

AND　OR

用語

- **変数**　x_1, x_2, x_3, x_4
- **リテラル**　$\underbrace{x_1, x_2, x_3, x_4}_{\text{正リテラル}}, \underbrace{\overline{x_1}, \overline{x_2}, \overline{x_3}, \overline{x_4}}_{\text{負リテラル}}$

変数への割当

$$\varphi = (x_1 \wedge \overline{x_2}) \vee (x_1 \vee \overline{x_3} \vee x_4)$$

$$\begin{aligned}
\varphi[x_1 = 0, x_2 = 0, x_3 = 1, x_4 = 1] &= (0 \wedge \overline{0}) \vee (0 \vee \overline{1} \vee 1) \\
&= (0 \wedge 1) \vee (0 \vee 0 \vee 1) \\
&= 0 \vee 1 = 1
\end{aligned}$$

割当

変数への割当

$$\varphi = (x_1 \wedge \overline{x_2}) \vee (x_1 \vee \overline{x_3} \vee x_4)$$

$$\begin{aligned}\varphi[x_1 = 0, x_2 = 0, x_3 = 1, x_4 = 1] &= (0 \wedge \overline{0}) \vee (0 \vee \overline{1} \vee 1) \\ &= (0 \wedge 1) \vee (0 \vee 0 \vee 1) \\ &= 0 \vee 1 = 1\end{aligned}$$

割当

$$\varphi[x_1 = 0] = (0 \wedge \overline{x_2}) \vee (0 \vee \overline{x_3} \vee x_4)$$

部分割当

$$= \overline{x_3} \vee x_4$$

変数への割当

$$\varphi = (x_1 \wedge \overline{x_2}) \vee (x_1 \vee \overline{x_3} \vee x_4)$$

$$\begin{aligned}\varphi[x_1 = 0, x_2 = 0, x_3 = 1, x_4 = 1] &= (0 \wedge \overline{0}) \vee (0 \vee \overline{1} \vee 1) \\ &= (0 \wedge 1) \vee (0 \vee 0 \vee 1) \\ &= 0 \vee 1 = 1\end{aligned}$$

割当

$$\varphi[x_1 = 0] = (0 \wedge \overline{x_2}) \vee (0 \vee \overline{x_3} \vee x_4)$$

部分割当

$$= \overline{x_3} \vee x_4$$

定義：充足割当 (satisfying assignment)

論理式 φ を 1 とする割当を φ の **充足割当** と呼ぶ

変数への割当

$$\varphi = (x_1 \wedge \overline{x_2}) \vee (x_1 \vee \overline{x_3} \vee x_4)$$

$$\begin{aligned}\varphi[x_1 = 0, x_2 = 0, x_3 = 1, x_4 = 1] &= (0 \wedge \overline{0}) \vee (0 \vee \overline{1} \vee 1) \\ &= (0 \wedge 1) \vee (0 \vee 0 \vee 1) \\ &= 0 \vee 1 = 1\end{aligned}$$

割当

$$\begin{aligned}\varphi[x_1 = 0] &= (0 \wedge \overline{x_2}) \vee (0 \vee \overline{x_3} \vee x_4) \\ &= \overline{x_3} \vee x_4\end{aligned}$$

部分割当

定義：充足可能 (satisfiable)

論理式 φ が **充足可能** であるとは,
φ を 1 とする割当 (充足割当) が存在すること

定義：連言標準形

論理式 φ が **連言標準形** で表されているとは,
φ が「リテラルの OR の AND」で書かれていること

$$\varphi = \underbrace{\underbrace{(x_1 \vee \overline{x_2})}_{\text{リテラルの OR}} \wedge \underbrace{(x_1 \vee \overline{x_3} \vee x_4)}_{\text{リテラルの OR}}}_{\text{リテラルの OR の AND}}$$

連言標準形：conjunctive normal form (CNF)

$\boxed{\text{用語}}$：節 (clause) $=$ リテラルの OR

節 C のサイズ $=$ C が含むリテラルの数

CNF-SAT

問題：CNF-SAT

入力： 連言標準形で表された論理式 φ
出力： φ が充足可能である $\Rightarrow$ Yes
φ が充足可能ではない $\Rightarrow$ No

$$\varphi = (x_1 \vee \overline{x_2}) \wedge (x_1 \vee \overline{x_3} \vee x_4)$$

注：CNF-SAT は NP 完全 (Cook '71; Levin '73)

問題：CNF-SAT

入力： 連言標準形で表された論理式 φ

出力： φ が充足可能である $\Rightarrow$ Yes

φ が充足可能ではない $\Rightarrow$ No

$$\varphi = (x_1 \vee \overline{x_2}) \wedge (x_1 \vee \overline{x_3} \vee x_4)$$

$$\begin{aligned} \varphi[x_1 = 1, x_2 = 1, x_3 = 0, x_4 = 0] &= (1 \vee \overline{1}) \wedge (1 \vee \overline{0} \vee 0) \\ &= 1 \end{aligned}$$

$\rightsquigarrow$ Yes

注：CNF-SAT は NP 完全 (Cook '71; Levin '73)

$k \geq 1$ は正整数

問題：k-SAT

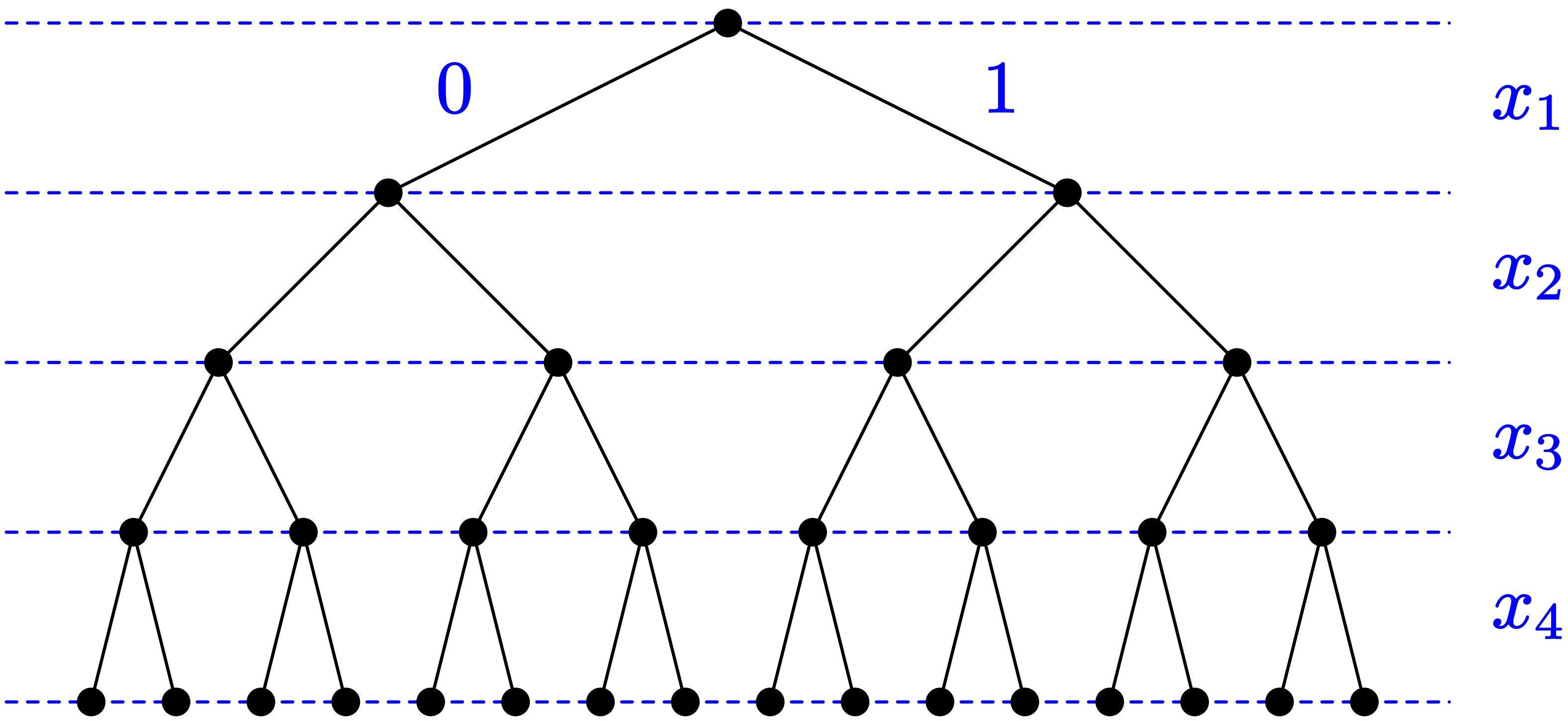
入力： 連言標準形で表された論理式 φ で，
各節のサイズが k 以下であるもの

出力： φ が充足可能である $\Rightarrow$ Yes
φ が充足可能ではない $\Rightarrow$ No

$$\varphi = (x_1 \vee \overline{x_2}) \wedge (x_1 \vee \overline{x_3} \vee x_4)$$

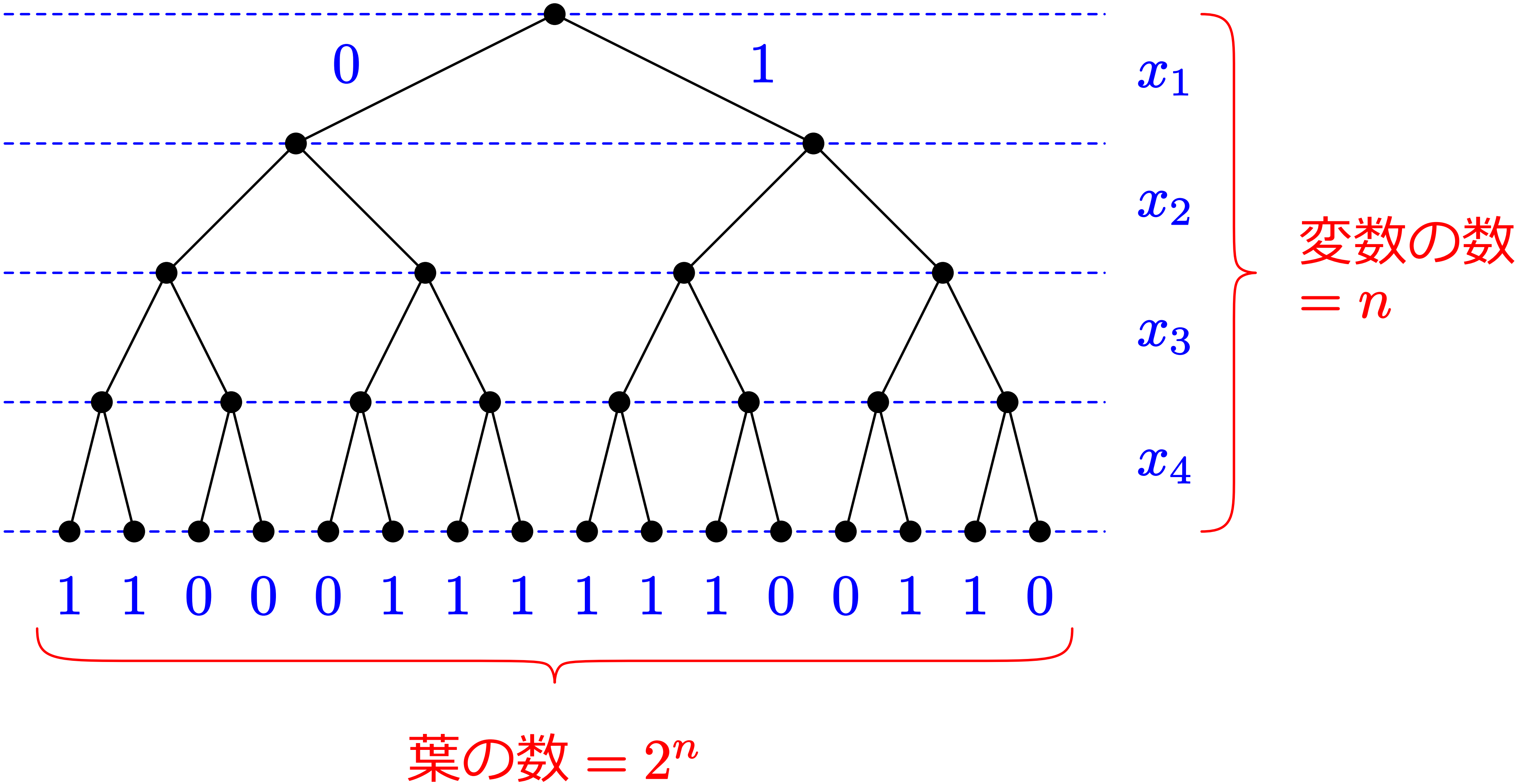
注：$k \geq 3$ のとき，k-SAT は NP 完全 (Karp '72)

$$\varphi = (x_1 \vee x_2 \vee \overline{x_3}) \wedge (\overline{x_2} \vee x_3 \vee x_4) \wedge (\overline{x_1} \vee \overline{x_3} \vee \overline{x_4})$$



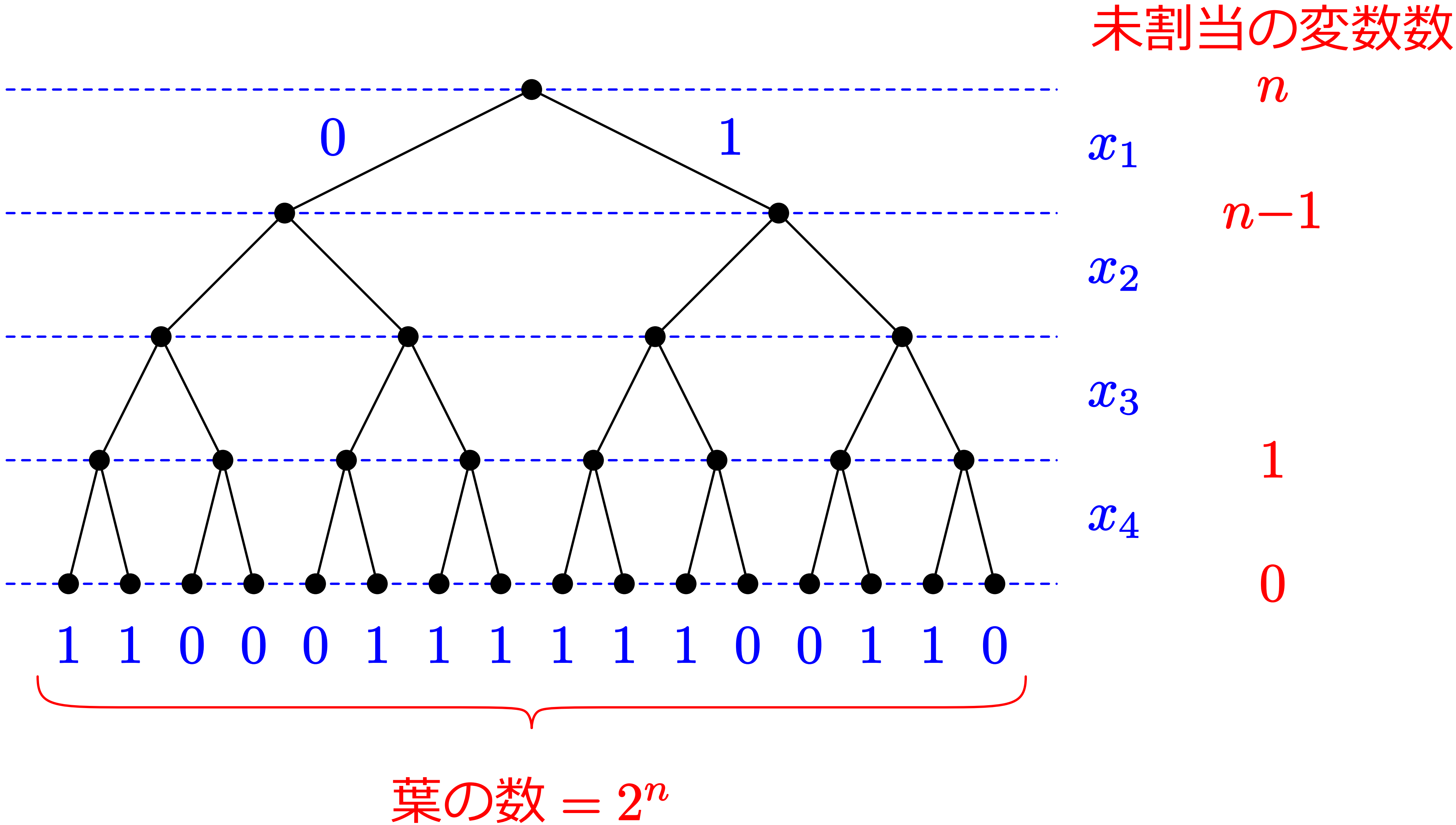
3-SAT を解く：しらみつぶし？

$$\varphi = (x_1 \lor x_2 \lor \overline{x_3}) \land (\overline{x_2} \lor x_3 \lor x_4) \land (\overline{x_1} \lor \overline{x_3} \lor \overline{x_4})$$



3-SAT を解く：しらみつぶし？

$$\varphi = (x_1 \vee x_2 \vee \overline{x_3}) \wedge (\overline{x_2} \vee x_3 \vee x_4) \wedge (\overline{x_1} \vee \overline{x_3} \vee \overline{x_4})$$



3-SAT：工夫

$$\varphi = \underline{(x_1 \vee x_2 \vee \overline{x_3})} \wedge (\overline{x_2} \vee x_3 \vee x_4) \wedge (\overline{x_1} \vee \overline{x_3} \vee \overline{x_4})$$

φ を 1 (真) にするには
この節も 1 (真) にしなくてはならない

3-SAT：工夫

$$\varphi = \underline{(x_1 \lor x_2 \lor \overline{x_3})} \land (\overline{x_2} \lor x_3 \lor x_4) \land (\overline{x_1} \lor \overline{x_3} \lor \overline{x_4})$$

φ を 1 (真) にするには
この節も 1 (真) にしなくてはならない

この節を 1 にする部分割当は？

x_1	x_2	x_3
0	0	0
0	0	1
0	1	0
0	1	1
1	0	0
1	0	1
1	1	0
1	1	1

$$\varphi = \underline{(x_1 \vee x_2 \vee \overline{x_3})} \wedge (\overline{x_2} \vee x_3 \vee x_4) \wedge (\overline{x_1} \vee \overline{x_3} \vee \overline{x_4})$$

φ を 1 (真) にするには
この節も 1 (真) にしなくてはならない

この節を 1 にする部分割当は？

x_1	x_2	x_3
0	0	0
~~0~~	~~0~~	~~1~~
0	1	0
0	1	1
1	0	0
1	0	1
1	1	0
1	1	1

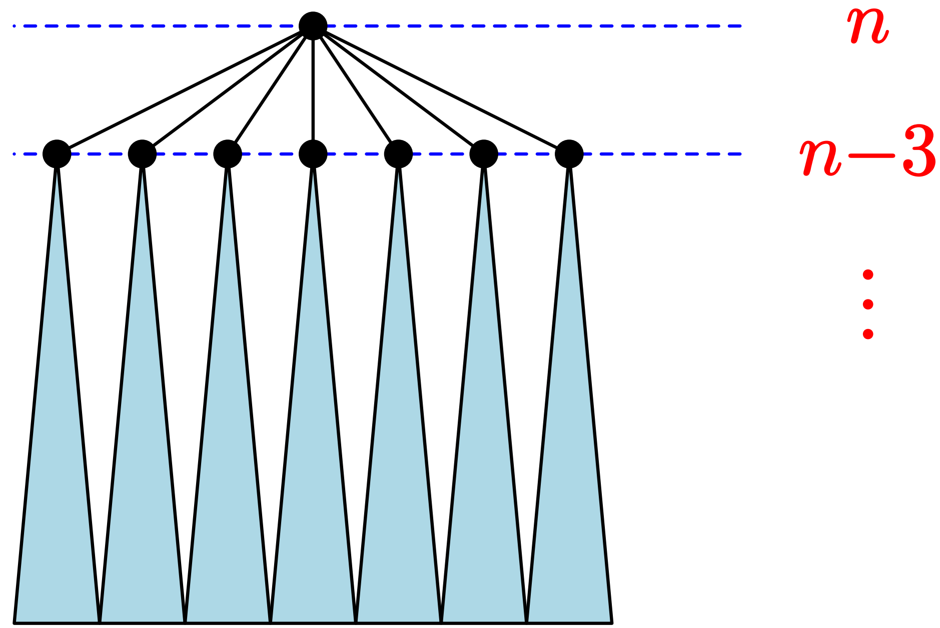
3-SAT：工夫

$$\varphi = \underline{(x_1 \vee x_2 \vee \overline{x_3})} \wedge (\overline{x_2} \vee x_3 \vee x_4) \wedge (\overline{x_1} \vee \overline{x_3} \vee \overline{x_4})$$

φ を 1 (真) にするには
この節も 1 (真) にしなくてはならない

この節を 1 にする部分割当は？

x_1	x_2	x_3
0	0	0
~~0~~	~~0~~	~~1~~
0	1	0
0	1	1
1	0	0
1	0	1
1	1	0
1	1	1



3-SAT：工夫

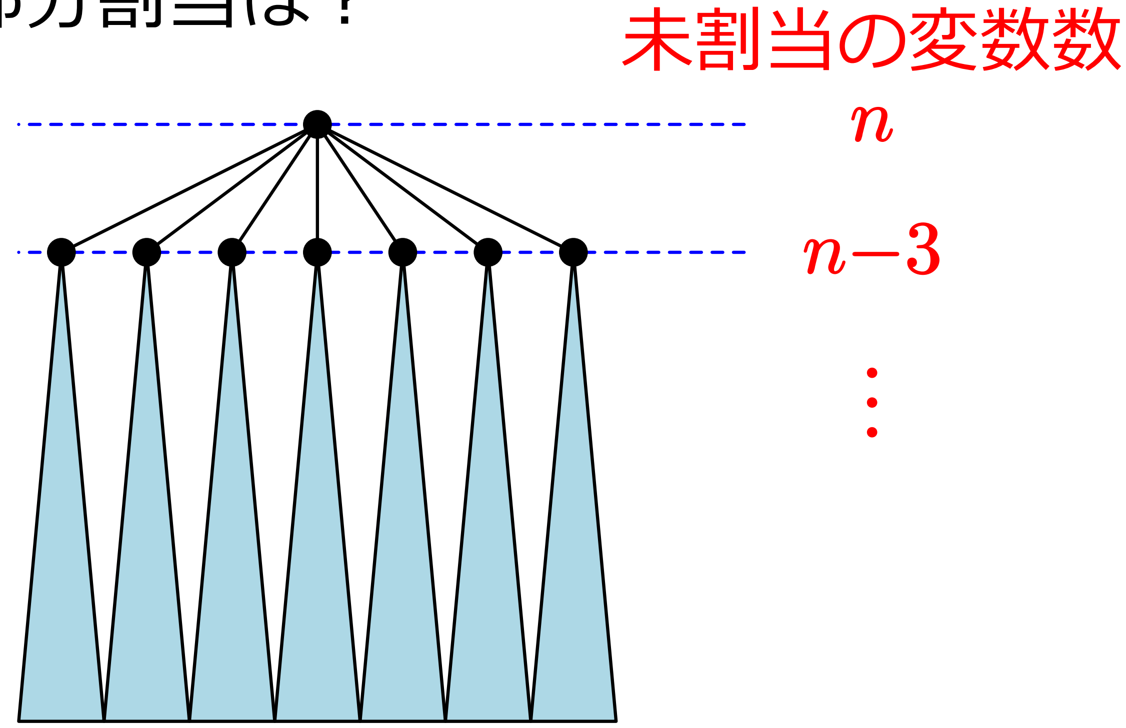
$$\varphi = \underline{(x_1 \vee x_2 \vee \overline{x_3})} \wedge \underline{(\overline{x_2} \vee x_3 \vee x_4)} \wedge (\overline{x_1} \vee \overline{x_3} \vee \overline{x_4})$$

φ を 1 (真) にするには

この節も 1 (真) にしなくてはならない

この節を 1 にする部分割当は？

x_1	x_2	x_3
0	0	0
~~0~~	~~0~~	~~1~~
0	1	0
0	1	1
1	0	0
1	0	1
1	1	0
1	1	1



性質

2-SAT は多項式時間で解ける

アルゴリズムのアイディアの説明は付録で

$$\varphi = (x_1 \vee x_2) \wedge (\overline{x_2} \vee x_3) \wedge (\overline{x_1} \vee \overline{x_3})$$

性質

2-SAT は多項式時間で解ける

アルゴリズムのアイディアの説明は付録で

$$\varphi = (x_1 \vee x_2) \wedge (\overline{x_2} \vee x_3) \wedge (\overline{x_1} \vee \overline{x_3})$$

サイズが 3 の節に現れる変数はすべて異なるとしてよい

$$x_1 \vee x_1 \vee x_2 = x_1 \vee x_2$$

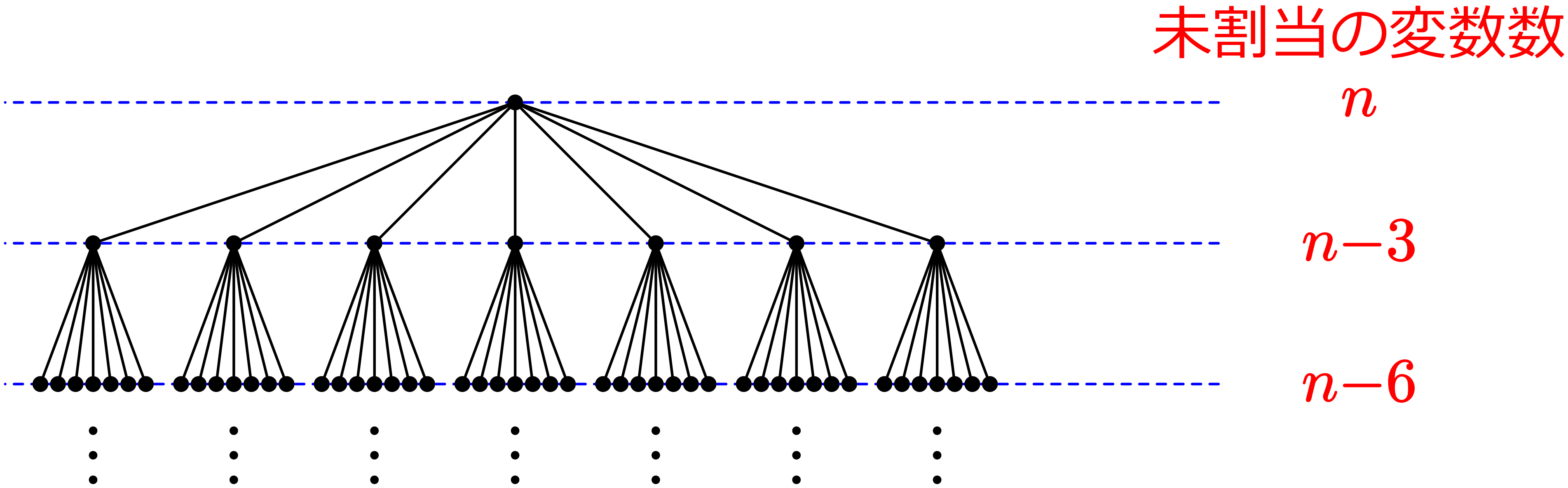
任意の割当において

$$x_1 \vee \overline{x_1} \vee x_2 = 1$$

<u>アルゴリズム B(φ)</u>

1. if φ がサイズ 3 の節を持たない
   - 2-SAT のアルゴリズムで解き，Yes/No を出力
2. $(\ell_1 \vee \ell_2 \vee \ell_3) = \varphi$ のサイズ 3 の節
3. 次のどれかが Yes を出力 $\Rightarrow$ Yes を出力
   次のすべてが No を出力 $\Rightarrow$ No を出力
   - B($\varphi[\ell_1 = 0, \ell_2 = 0, \ell_3 = 1]$)
   - B($\varphi[\ell_1 = 0, \ell_2 = 1, \ell_3 = 0]$)
   - B($\varphi[\ell_1 = 0, \ell_2 = 1, \ell_3 = 1]$)
   - B($\varphi[\ell_1 = 1, \ell_2 = 0, \ell_3 = 0]$)
   - B($\varphi[\ell_1 = 1, \ell_2 = 0, \ell_3 = 1]$)
   - B($\varphi[\ell_1 = 1, \ell_2 = 1, \ell_3 = 0]$)
   - B($\varphi[\ell_1 = 1, \ell_2 = 1, \ell_3 = 1]$)

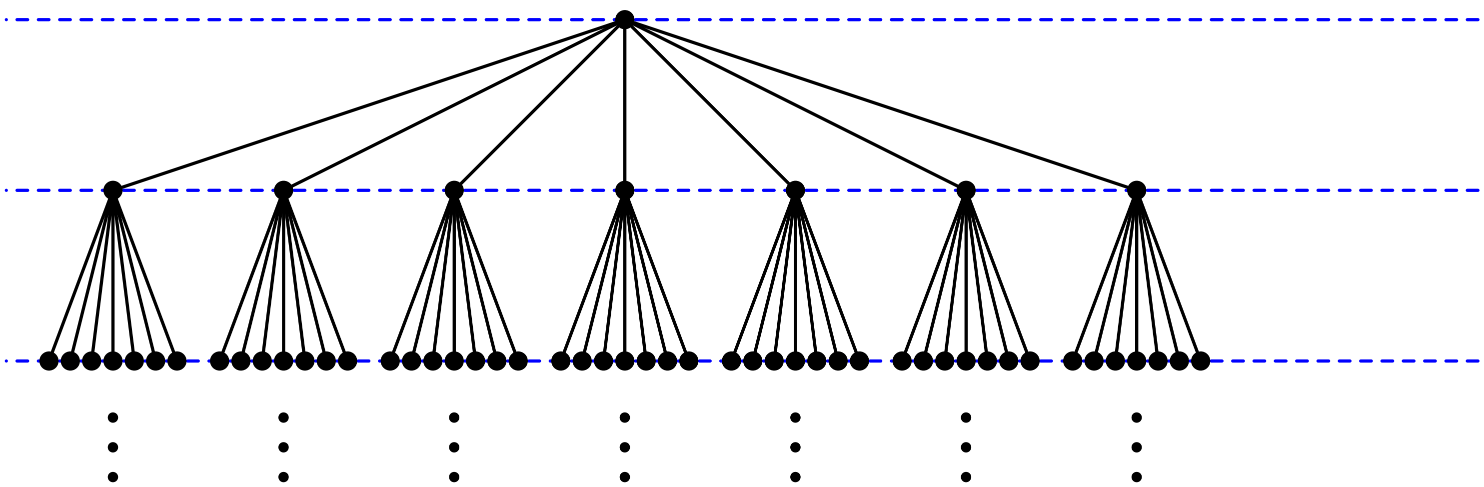
アルゴリズム B：探索木



$T(n) =$ 変数数 n の論理式を入力としたときの
探索木の葉の数 (の最大値)

$$T(n) \leq 7 \cdot T(n-3)$$

アルゴリズム B：探索木



$T(n) =$ 変数数 n の論理式を入力としたときの
探索木の葉の数 (の最大値)

$$T(n) \leq 7 \cdot T(n-3)$$

特性方程式 $x^3 = 7 \quad \rightsquigarrow \quad$ 正の実数解は 1.9130

$$\therefore \quad T(n) = \boxed{O(1.9130^n)}$$

$$\varphi = \underline{(x_1 \vee x_2 \vee \overline{x_3})} \wedge (\overline{x_2} \vee x_3 \vee x_4) \wedge (\overline{x_1} \vee \overline{x_3} \vee \overline{x_4})$$

φ を 1 (真) にするには
この節も 1 (真) にしなくてはならない

この節を 1 にする部分割当は？

x_1	x_2	$\overline{x_3}$
~~0~~	~~0~~	~~0~~
0	0	1
0	1	0
0	1	1
1	0	0
1	0	1
1	1	0
1	1	1

$$\varphi = \underline{(x_1 \vee x_2 \vee \overline{x_3})} \wedge (\overline{x_2} \vee x_3 \vee x_4) \wedge (\overline{x_1} \vee \overline{x_3} \vee \overline{x_4})$$

φ を 1 (真) にするには
この節も 1 (真) にしなくてはならない

この節を 1 にする部分割当は？

x_1	x_2	$\overline{x_3}$
~~0~~	~~0~~	~~0~~
0	0	1
0	1	0
0	1	1
1	0	0
1	0	1
1	1	0
1	1	1

x_1	x_2	$\overline{x_3}$
~~0~~	~~0~~	~~0~~
0	0	1
0	1	
1		

3-SAT：さらに工夫

$$\varphi = \underline{(x_1 \vee x_2 \vee \overline{x_3})} \wedge (\overline{x_2} \vee x_3 \vee x_4) \wedge (\overline{x_1} \vee \overline{x_3} \vee \overline{x_4})$$

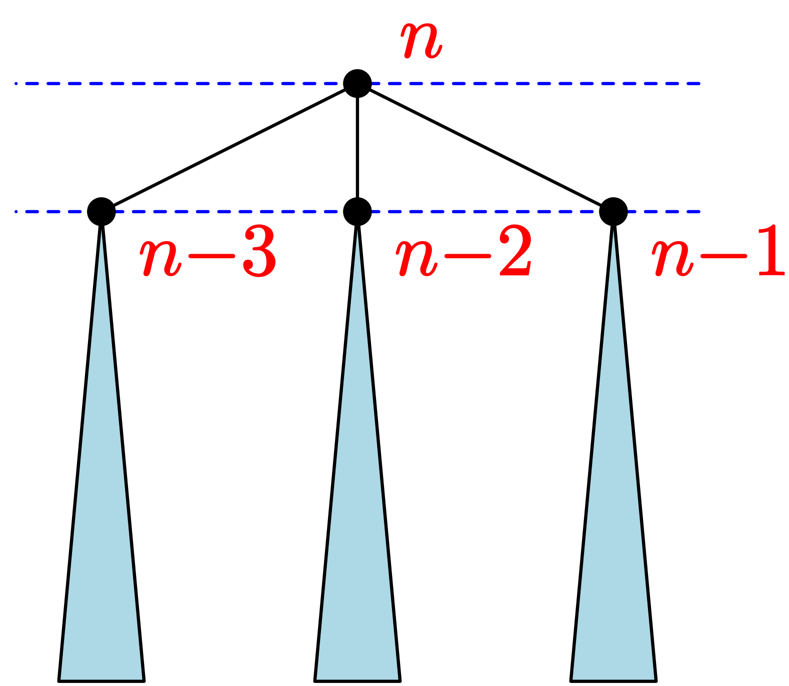
φ を 1 (真) にするには
この節も 1 (真) にしなくてはならない

この節を 1 にする部分割当は？

x_1	x_2	$\overline{x_3}$
~~0~~	~~0~~	~~0~~
0	0	1
0	1	0
0	1	1
1	0	0
1	0	1
1	1	0
1	1	1

x_1	x_2	$\overline{x_3}$
~~0~~	~~0~~	~~0~~
0	0	1
0	1	
1		

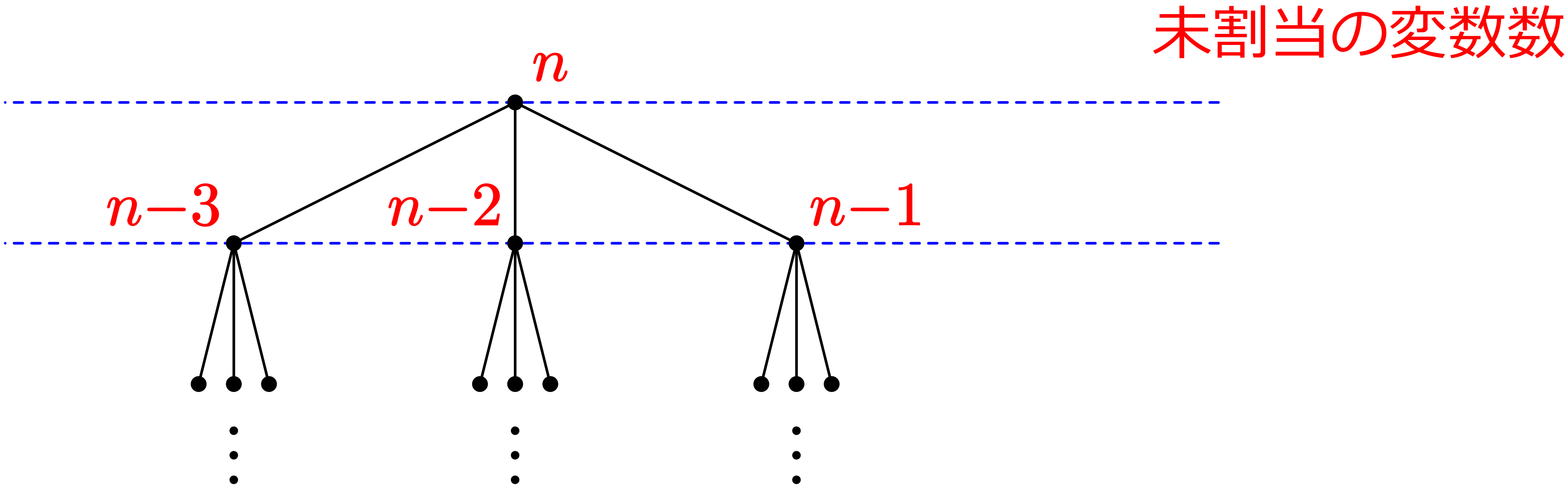
未割当の変数数



アルゴリズム C(φ)

1. if φ がサイズ 3 の節を持たない
   - 2-SAT のアルゴリズムで解き，Yes/No を出力
2. $(\ell_1 \vee \ell_2 \vee \ell_3) = \varphi$ のサイズ 3 の節
3. 次のどれかが Yes を出力 $\Rightarrow$ Yes を出力
   次のすべてが No を出力 $\Rightarrow$ No を出力
   - C($\varphi[\ell_1 = 0, \ell_2 = 0, \ell_3 = 1]$)
   - C($\varphi[\ell_1 = 0, \ell_2 = 1]$)
   - C($\varphi[\ell_1 = 1]$)

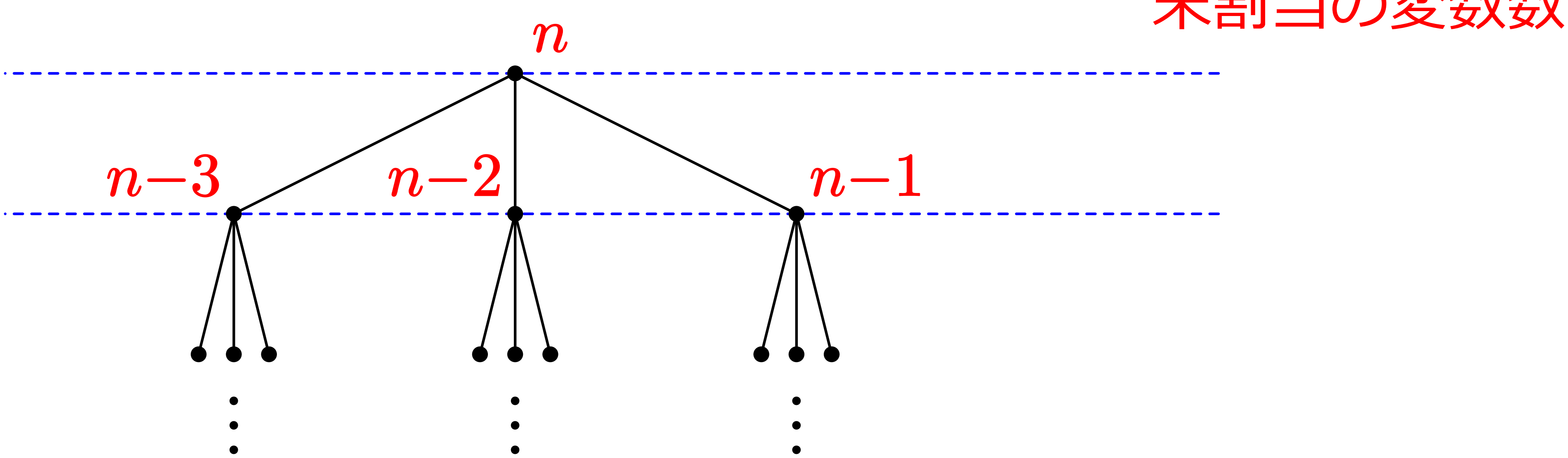
アルゴリズムC：探索木



$T(n) =$ 変数数 n の論理式を入力としたときの
探索木の葉の数 (の最大値)

$$T(n) \leq T(n-3) + T(n-2) + T(n-1)$$

アルゴリズム C：探索木



$T(n) =$ 変数数 n の論理式を入力としたときの
探索木の葉の数 (の最大値)

$$T(n) \leq T(n-3) + T(n-2) + T(n-1)$$

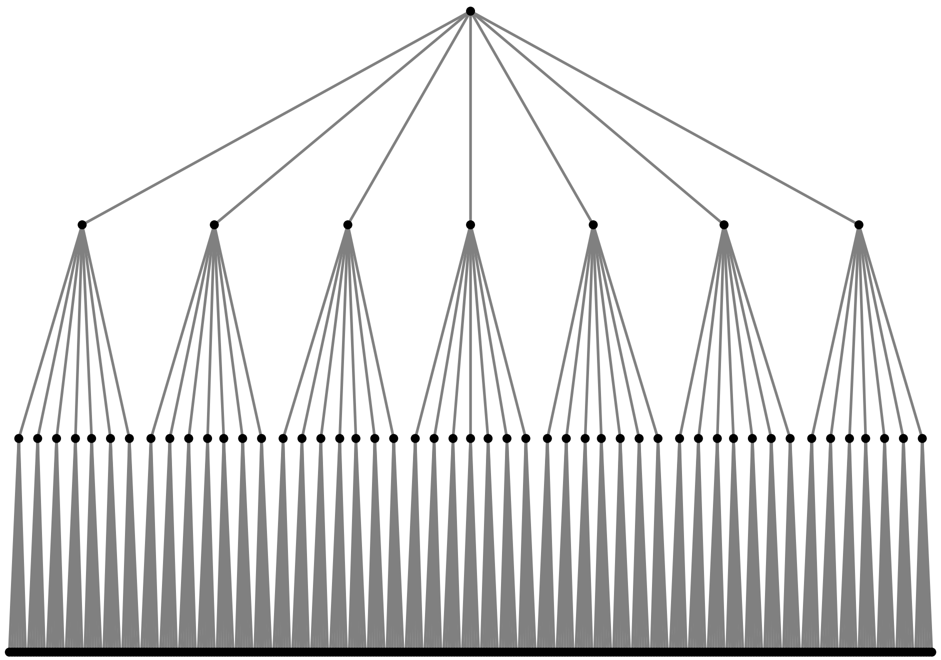
特性方程式 $x^3 = 1 + x + x^2 \quad \rightsquigarrow \quad$ 正の実数解は 1.8393

$$\therefore \quad T(n) = \boxed{O(1.8393^n)}$$

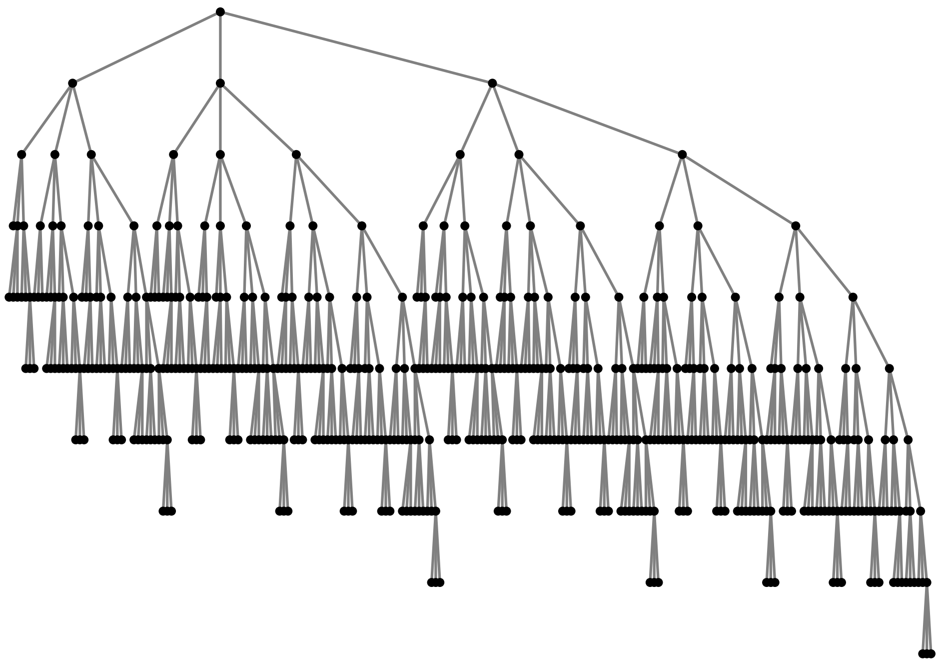
注：アルゴリズム B は $O(1.9130^n)$

比較：アルゴリズム B と C

アルゴリズム B



アルゴリズム C



次回の予告

第 2 回以降 3 回

分枝アルゴリズム (branching algorithm) の設計と解析

第 2 回 (今回)

- 分枝アルゴリズムの基礎

第 3 回 (次回)

- 分枝アルゴリズムの改良法

第 4 回

- 測度統治法 (measure & conquer) による改良

本日の内容

1. 分枝アルゴリズムの設計：基礎
2. 分枝アルゴリズムの解析：基礎
3. 他の例：充足可能性問題
4. **付録：2-SAT のアルゴリズム**

2-SAT のアルゴリズム

性質

2-SAT は多項式時間で解ける

$$\varphi = (x_1 \vee x_2) \wedge (\overline{x_2} \vee x_3) \wedge (\overline{x_1} \vee \overline{x_3})$$

多項式時間アルゴリズムはいくつか知られているが，
その中の 1 つを紹介する

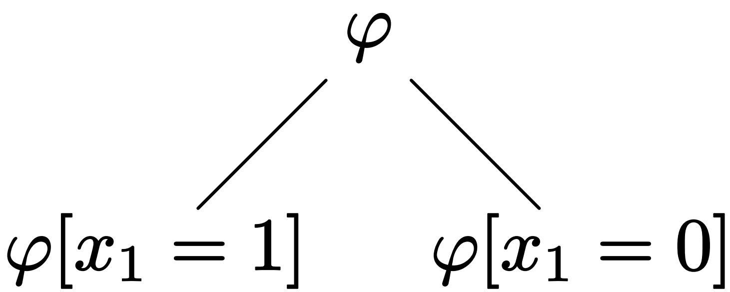
- **単位伝播** に基づくもの

ポイント

前処理 の威力

2-SAT：例 (1)

$$\varphi = (x_1 \vee x_2) \wedge (\overline{x_1} \vee x_3) \wedge (\overline{x_3} \vee x_4) \wedge (\overline{x_3} \vee x_5)$$
$$\wedge (\overline{x_4} \vee \overline{x_5}) \wedge (\overline{x_5} \vee x_6) \wedge (\overline{x_6} \vee x_7) \wedge (\overline{x_2} \vee \overline{x_7})$$



2-SAT：例 (1)

$$
\begin{aligned}
\varphi = {} & (x_1 \lor x_2) \land (\overline{x_1} \lor x_3) \land (\overline{x_3} \lor x_4) \land (\overline{x_3} \lor x_5) \\
& \land (\overline{x_4} \lor \overline{x_5}) \land (\overline{x_5} \lor x_6) \land (\overline{x_6} \lor x_7) \land (\overline{x_2} \lor \overline{x_7})
\end{aligned}
$$

$$
\begin{array}{ccc}
 & \varphi & \\
\swarrow & & \searrow \\
\varphi[x_1 = 1] & & \varphi[x_1 = 0]
\end{array}
$$

$$
\begin{aligned}
\varphi[x_1 = 1] &= (1 \lor x_2) \land (0 \lor x_3) \land (\overline{x_3} \lor x_4) \land (\overline{x_3} \lor x_5) \\
&\quad \land (\overline{x_4} \lor \overline{x_5}) \land (\overline{x_5} \lor x_6) \land (\overline{x_6} \lor x_7) \land (\overline{x_2} \lor \overline{x_7}) \\
&= (x_3) \land (\overline{x_3} \lor x_4) \land (\overline{x_3} \lor x_5) \\
&\quad \land (\overline{x_4} \lor \overline{x_5}) \land (\overline{x_5} \lor x_6) \land (\overline{x_6} \lor x_7) \land (\overline{x_2} \lor \overline{x_7})
\end{aligned}
$$

2-SAT：例 (2)

$$\begin{aligned}\varphi = (x_1 \vee x_2) \wedge (\overline{x_1} \vee x_3) \wedge (\overline{x_3} \vee x_4) \wedge (\overline{x_3} \vee x_5) \\ \wedge (\overline{x_4} \vee \overline{x_5}) \wedge (\overline{x_5} \vee x_6) \wedge (\overline{x_6} \vee x_7) \wedge (\overline{x_2} \vee \overline{x_7})\end{aligned}$$

$$\varphi$$

$$\varphi[x_1 = 1] \qquad \varphi[x_1 = 0]$$

$$\begin{aligned}\varphi[x_1 = 1, x_3 = 1] &= (1) \wedge (0 \vee x_4) \wedge (0 \vee x_5) \wedge (\overline{x_4} \vee \overline{x_5}) \\ &\quad \wedge (\overline{x_5} \vee x_6) \wedge (\overline{x_6} \vee x_7) \wedge (\overline{x_2} \vee \overline{x_7}) \\ &= (x_4) \wedge (x_5) \wedge (\overline{x_4} \vee \overline{x_5}) \wedge \cdots\end{aligned}$$

2-SAT：例 (2)

$$\varphi = (x_1 \lor x_2) \land (\overline{x_1} \lor x_3) \land (\overline{x_3} \lor x_4) \land (\overline{x_3} \lor x_5) \\ \land (\overline{x_4} \lor \overline{x_5}) \land (\overline{x_5} \lor x_6) \land (\overline{x_6} \lor x_7) \land (\overline{x_2} \lor \overline{x_7})$$

$$\begin{array}{ccc} & \varphi & \\ \swarrow & & \searrow \\ \varphi[x_1 = 1] & & \varphi[x_1 = 0] \end{array}$$

$$\begin{aligned} \varphi[x_1 = 1, x_3 = 1] &= (1) \land (0 \lor x_4) \land (0 \lor x_5) \land (\overline{x_4} \lor \overline{x_5}) \\ &\quad \land (\overline{x_5} \lor x_6) \land (\overline{x_6} \lor x_7) \land (\overline{x_2} \lor \overline{x_7}) \\ &= (x_4) \land (x_5) \land (\overline{x_4} \lor \overline{x_5}) \land \cdots \end{aligned}$$

$$\begin{aligned} \varphi[x_1 = 1, x_3 = 1, x_4 = 1, x_5 = 1] &= (1) \land (1) \land (0 \lor 0) \land \cdots \\ &= (0) \land \cdots \end{aligned}$$

2-SAT：例 (2)

$$\begin{aligned}\varphi = (x_1 \vee x_2) \wedge (\overline{x_1} \vee x_3) \wedge (\overline{x_3} \vee x_4) \wedge (\overline{x_3} \vee x_5) \\ \wedge (\overline{x_4} \vee \overline{x_5}) \wedge (\overline{x_5} \vee x_6) \wedge (\overline{x_6} \vee x_7) \wedge (\overline{x_2} \vee \overline{x_7})\end{aligned}$$

$$\begin{tikzcd} & \varphi \arrow[dl, dash] \arrow[dr, dash] & \\ \varphi[x_1 = 1] & & \varphi[x_1 = 0] \\ 0 & & \end{tikzcd}$$

$$\begin{aligned}\varphi[x_1 = 1, x_3 = 1] &= (1) \wedge (0 \vee x_4) \wedge (0 \vee x_5) \wedge (\overline{x_4} \vee \overline{x_5}) \\ &\quad \wedge (\overline{x_5} \vee x_6) \wedge (\overline{x_6} \vee x_7) \wedge (\overline{x_2} \vee \overline{x_7}) \\ &= (x_4) \wedge (x_5) \wedge (\overline{x_4} \vee \overline{x_5}) \wedge \cdots\end{aligned}$$

$$\begin{aligned}\varphi[x_1 = 1, x_3 = 1, x_4 = 1, x_5 = 1] &= (1) \wedge (1) \wedge (0 \vee 0) \wedge \cdots \\ &= (0) \wedge \cdots\end{aligned}$$

2-SAT：例 (3)

$$\varphi = (x_1 \lor x_2) \land (\overline{x_1} \lor x_3) \land (\overline{x_3} \lor x_4) \land (\overline{x_3} \lor x_5)$$
$$\land (\overline{x_4} \lor \overline{x_5}) \land (\overline{x_5} \lor x_6) \land (\overline{x_6} \lor x_7) \land (\overline{x_2} \lor \overline{x_7})$$

$$\varphi$$

$$\varphi[x_1 = 1] \qquad \varphi[x_1 = 0]$$

$$0$$

$$\begin{aligned}
\varphi[x_1 = 0] &= (0 \lor x_2) \land (1 \lor x_3) \land (\overline{x_3} \lor x_4) \land (\overline{x_3} \lor x_5) \\
&\quad \land (\overline{x_4} \lor \overline{x_5}) \land (\overline{x_5} \lor x_6) \land (\overline{x_6} \lor x_7) \land (\overline{x_2} \lor \overline{x_7}) \\
&= (x_2) \land (\overline{x_3} \lor x_4) \land (\overline{x_3} \lor x_5) \\
&\quad \land (\overline{x_4} \lor \overline{x_5}) \land (\overline{x_5} \lor x_6) \land (\overline{x_6} \lor x_7) \land (\overline{x_2} \lor \overline{x_7})
\end{aligned}$$

2-SAT：例 (3)

$$\varphi = (x_1 \vee x_2) \wedge (\overline{x_1} \vee x_3) \wedge (\overline{x_3} \vee x_4) \wedge (\overline{x_3} \vee x_5)$$
$$\wedge (\overline{x_4} \vee \overline{x_5}) \wedge (\overline{x_5} \vee x_6) \wedge (\overline{x_6} \vee x_7) \wedge (\overline{x_2} \vee \overline{x_7})$$

$$\begin{array}{ccc} & \varphi & \\ \swarrow & & \searrow \\ \varphi[x_1 = 1] & & \varphi[x_1 = 0] \\ {\color{blue}0} & & \end{array}$$

$$\varphi[x_1 = 0] = (0 \vee x_2) \wedge (1 \vee x_3) \wedge (\overline{x_3} \vee x_4) \wedge (\overline{x_3} \vee x_5)$$
$$\wedge (\overline{x_4} \vee \overline{x_5}) \wedge (\overline{x_5} \vee x_6) \wedge (\overline{x_6} \vee x_7) \wedge (\overline{x_2} \vee \overline{x_7})$$
$$= (x_2) \wedge (\overline{x_3} \vee x_4) \wedge (\overline{x_3} \vee x_5)$$
$$\wedge (\overline{x_4} \vee \overline{x_5}) \wedge (\overline{x_5} \vee x_6) \wedge (\overline{x_6} \vee x_7) \wedge (\overline{x_2} \vee \overline{x_7})$$

$$\varphi[x_1 = 0, x_2 = 1] = (1) \wedge (\overline{x_3} \vee x_4) \wedge (\overline{x_3} \vee x_5) \wedge (\overline{x_4} \vee \overline{x_5})$$
$$\wedge (\overline{x_5} \vee x_6) \wedge (\overline{x_6} \vee x_7) \wedge (0 \vee \overline{x_7})$$

2-SAT：例 (4)

$$\varphi[x_1 = 0, x_2 = 1, x_7 = 0]$$
$$= (\overline{x_3} \vee x_4) \wedge (\overline{x_3} \vee x_5) \wedge (\overline{x_4} \vee \overline{x_5}) \wedge (\overline{x_5} \vee x_6) \wedge (\overline{x_6} \vee 0) \wedge (1)$$
$$= (\overline{x_3} \vee x_4) \wedge (\overline{x_3} \vee x_5) \wedge (\overline{x_4} \vee \overline{x_5}) \wedge (\overline{x_5} \vee x_6) \wedge (\overline{x_6})$$

$$\varphi[x_1 = 0, x_2 = 1, x_7 = 0]$$
$$= (\overline{x_3} \vee x_4) \wedge (\overline{x_3} \vee x_5) \wedge (\overline{x_4} \vee \overline{x_5}) \wedge (\overline{x_5} \vee x_6) \wedge (\overline{x_6} \vee 0) \wedge (1)$$
$$= (\overline{x_3} \vee x_4) \wedge (\overline{x_3} \vee x_5) \wedge (\overline{x_4} \vee \overline{x_5}) \wedge (\overline{x_5} \vee x_6) \wedge (\overline{x_6})$$

$$\varphi[x_1 = 0, x_2 = 1, x_7 = 0, x_6 = 0]$$
$$= (\overline{x_3} \vee x_4) \wedge (\overline{x_3} \vee x_5) \wedge (\overline{x_4} \vee \overline{x_5}) \wedge (\overline{x_5} \vee 0) \wedge (1)$$
$$= (\overline{x_3} \vee x_4) \wedge (\overline{x_3} \vee x_5) \wedge (\overline{x_4} \vee \overline{x_5}) \wedge (\overline{x_5})$$

2-SAT：例 (4)

$$
\begin{aligned}
&\varphi[x_1 = 0, x_2 = 1, x_7 = 0] \\
&\quad = (\overline{x_3} \vee x_4) \wedge (\overline{x_3} \vee x_5) \wedge (\overline{x_4} \vee \overline{x_5}) \wedge (\overline{x_5} \vee x_6) \wedge (\overline{x_6} \vee 0) \wedge (1) \\
&\quad = (\overline{x_3} \vee x_4) \wedge (\overline{x_3} \vee x_5) \wedge (\overline{x_4} \vee \overline{x_5}) \wedge (\overline{x_5} \vee x_6) \wedge (\overline{x_6})
\end{aligned}
$$

$$
\begin{aligned}
&\varphi[x_1 = 0, x_2 = 1, x_7 = 0, x_6 = 0] \\
&\quad = (\overline{x_3} \vee x_4) \wedge (\overline{x_3} \vee x_5) \wedge (\overline{x_4} \vee \overline{x_5}) \wedge (\overline{x_5} \vee 0) \wedge (1) \\
&\quad = (\overline{x_3} \vee x_4) \wedge (\overline{x_3} \vee x_5) \wedge (\overline{x_4} \vee \overline{x_5}) \wedge (\overline{x_5})
\end{aligned}
$$

$$
\begin{aligned}
&\varphi[x_1 = 0, x_2 = 1, x_7 = 0, x_6 = 0, x_5 = 0] \\
&\quad = (\overline{x_3} \vee x_4) \wedge (\overline{x_3} \vee 0) \wedge (\overline{x_4} \vee 1) \wedge (1) \\
&\quad = (\overline{x_3} \vee x_4) \wedge (\overline{x_3})
\end{aligned}
$$

$$\varphi[x_1 = 0, x_2 = 1, x_7 = 0]$$
$$= (\overline{x_3} \vee x_4) \wedge (\overline{x_3} \vee x_5) \wedge (\overline{x_4} \vee \overline{x_5}) \wedge (\overline{x_5} \vee x_6) \wedge (\overline{x_6} \vee 0) \wedge (1)$$
$$= (\overline{x_3} \vee x_4) \wedge (\overline{x_3} \vee x_5) \wedge (\overline{x_4} \vee \overline{x_5}) \wedge (\overline{x_5} \vee x_6) \wedge (\overline{x_6})$$

$$\varphi[x_1 = 0, x_2 = 1, x_7 = 0, x_6 = 0]$$
$$= (\overline{x_3} \vee x_4) \wedge (\overline{x_3} \vee x_5) \wedge (\overline{x_4} \vee \overline{x_5}) \wedge (\overline{x_5} \vee 0) \wedge (1)$$
$$= (\overline{x_3} \vee x_4) \wedge (\overline{x_3} \vee x_5) \wedge (\overline{x_4} \vee \overline{x_5}) \wedge (\overline{x_5})$$

$$\varphi[x_1 = 0, x_2 = 1, x_7 = 0, x_6 = 0, x_5 = 0]$$
$$= (\overline{x_3} \vee x_4) \wedge (\overline{x_3} \vee 0) \wedge (\overline{x_4} \vee 1) \wedge (1)$$
$$= (\overline{x_3} \vee x_4) \wedge (\overline{x_3})$$

$$\varphi[x_1 = 0, x_2 = 1, x_7 = 0, x_6 = 0, x_5 = 0, x_3 = 0]$$
$$= (1 \vee x_4) \wedge (1) = 1$$

性質：単位節

CNF 論理式 φ がサイズ 1 の節 (ℓ) を持つとき，次は同値

- φ が充足可能
- $\varphi[\ell = 1]$ が充足可能

アルゴリズム：単位伝播(φ) // φ は CNF 論理式

1. φ がサイズ 1 の節 (ℓ) を持つ限り反復
   - $\varphi = \varphi[\ell = 1]$
2. φ を出力

性質：φ が充足可能 $\Leftrightarrow$ 単位伝播(φ) が充足可能

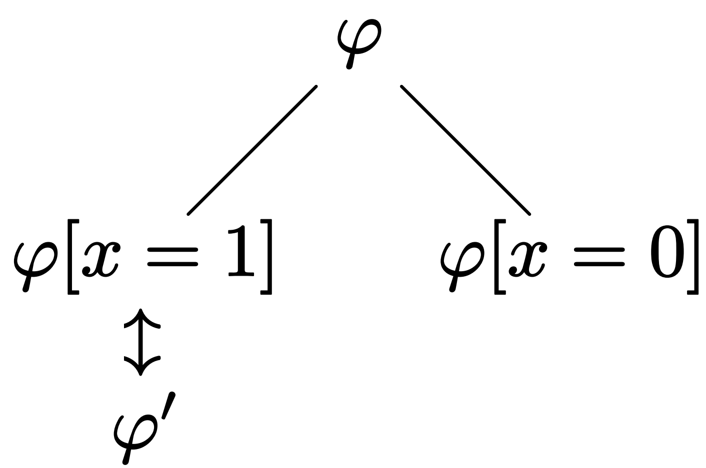
2-SAT：単位伝播の性質

φ：CNF 論理式で，すべての節のサイズが 2 以下のもの

φ'：単位伝播$(\varphi[x=1])$　(ただし，x は φ に現れる変数)

性質：2-SAT と単位伝播

1. φ' が節 (0) を持つとき
   φ が充足可能 $\Leftrightarrow$ $\varphi[x=0]$ が充足可能
2. φ' が節 (0) を持たないとき
   φ が充足可能 $\Leftrightarrow$ φ' が充足可能



<u>アルゴリズム</u>：2-SAT(φ) // φ は 2-CNF 論理式

1. if φ が節 (0) を持つ
   No を出力
2. if φ のすべての節が (1)
   Yes を出力
3. $x = \varphi$ に現れる変数
4. $\varphi' =$ 単位伝搬($\varphi[x = 1]$)
5. if φ' が節 (0) を持つ
   2-SAT($\varphi[x = 0]$) を実行
6. if φ が節 (0) を持たない
   2-SAT(φ') を実行

正しさ と 多項式時間で動くことは すぐにわかる

(前のページの性質から)

2-SAT：性質の証明 (1)

φ：CNF 論理式で，すべての節のサイズが 2 以下のもの

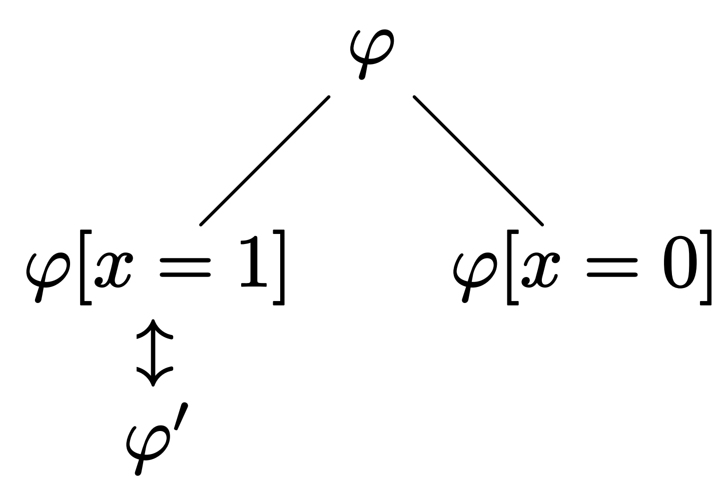
φ'：単位伝播($\varphi[x=1]$)　(ただし，x は φ に現れる変数)

性質：2-SAT と単位伝播

1. φ' が節 (0) を持つとき
   φ が充足可能 $\Leftrightarrow$ $\varphi[x=0]$ が充足可能

証明：φ' が節 (0) を持つとする

- φ' は充足可能ではない
- $\therefore \varphi[x=1]$ は充足可能ではない



- $\therefore \varphi$ が充足可能 $\Leftrightarrow$ $\varphi[x=1]$ か $\varphi[x=0]$ が充足可能
  $\Leftrightarrow$ $\varphi[x=0]$ が充足可能　□

2-SAT：性質の証明 (2-1)

φ：CNF 論理式で，すべての節のサイズが 2 以下のもの

φ'：単位伝播$(\varphi[x=1])$　(ただし，x は φ に現れる変数)

性質：2-SAT と単位伝播

2. φ' が節 $()$ を持たないとき
   φ が充足可能 $\Leftrightarrow$ φ' が充足可能

<u>証明</u>：φ' が節 $()$ を持たないとする

- φ' のすべての節のサイズは 2 か 0 ($\because$ 単位伝播)
- $\therefore \varphi'$ のサイズ 2 の節はどれも φ の節

2-SAT：性質の証明 (2-1)

φ：CNF 論理式で，すべての節のサイズが 2 以下のもの

φ'：単位伝播$(\varphi[x=1])$　(ただし，x は φ に現れる変数)

性質：2-SAT と単位伝播

2. φ' が節 (0) を持たないとき
   φ が充足可能 $\Leftrightarrow$ φ' が充足可能

<u>証明</u>：φ' が節 (0) を持たないとする

- φ' のすべての節のサイズは 2 か 0 ($\because$ 単位伝播)
- $\therefore \varphi'$ のサイズ 2 の節はどれも φ の節

[$\Rightarrow$]：φ が充足可能だと仮定する

- φ を 1 にする割当 α が存在
- α は φ のすべての節を 1 にする
- $\therefore \alpha$ は φ' のサイズ 2 の節をすべて 1 にする
- $\therefore \alpha$ は φ' を 1 にする ($\because \varphi'$ は (0) を持たない)　$\square$

2-SAT：性質の証明 (2-2)

φ：CNF 論理式で，すべての節のサイズが 2 以下のもの

φ'：単位伝播$(\varphi[x = 1])$　(ただし，x は φ に現れる変数)

性質：2-SAT と単位伝播

2. φ' が節 $()$ を持たないとき
   φ が充足可能 $\Leftrightarrow$ φ' が充足可能

<u>証明</u>：

[$\Leftarrow$]：φ' が充足可能だと仮定する

- 「φ が充足可能 $\Leftrightarrow$ $\varphi[x = 1]$ か $\varphi[x = 0]$ が充足可能」で，「$\varphi[x = 1]$ が充足可能 $\Leftrightarrow$ 単位伝播$(\varphi[x = 1])$ が充足可能」なので，φ も充足可能 $\square$