

2 木・閉路と探索

第 1 回ではグラフの基本概念(頂点・辺・次数・連結性・連結成分)を学んだ. 今回では, 連結で「閉路をもたない」グラフである **木** (tree) を主役として, 閉路と木のもつ著しい性質を調べ, さらにグラフの全頂点を組織的にたどる **探索** (search) の方法(幅優先・深さ優先)を学ぶ. 木はデータ構造や通信網の設計, 場合の数の数え上げなど至るところに現れる, 最も基本的なグラフの一つである.

● 2-1 : 道と閉路

第 1 回で, 頂点と辺が交互に並んだ **歩道** (walk) を導入した. 復習すると, 歩道とは列 $v_0, e_1, v_1, \dots, e_k, v_k$ であって, 各 i について $\psi(e_i) = \{v_{i-1}, v_i\}$ を満たすものであった. その特別な場合として, 次の道と閉路を定義する.

定義 2.1. グラフ $G = (V, E, \psi)$ における歩道 $v_0, e_1, v_1, \dots, e_k, v_k$ について,

- (a) 頂点 $v_0, v_1, \dots, v_k$ がすべて相異なるとき, この歩道を v_0 から v_k への **道** (path) という.
- (b) $k \geq 1$ かつ $v_0 = v_k$ であって, 辺 $e_1, \dots, e_k$ がすべて相異なり, かつ頂点 $v_0, v_1, \dots, v_{k-1}$ がすべて相異なるとき, この歩道を長さ k の **閉路** (cycle) という.

多重グラフでは, 閉路の長さ k は小さな値も取り得る. $k = 1$ の閉路は $\psi(e_1) = \{v_0\}$, すなわちループ 1 本である. $k = 2$ の閉路は, 同じ 2 頂点を結ぶ相異なる 2 本の辺, すなわち多重辺である. $k \geq 3$ の閉路は, 相異なる頂点を順にたどって出発点へ戻る通常の閉路である.

定義 2.2. 閉路を 1 つももたないグラフを **非巡回的** (acyclic) であるという.

上の観察から, ループは長さ 1 の閉路, 多重辺は長さ 2 の閉路であった. ゆえに非巡回的なグラフはループも多重辺ももたない. すなわち, **非巡回的なグラフは単純グラフである**.

命題 2.3. グラフ G において u から v への歩道が存在するならば, u から v への道が存在する.

証明. u から v への歩道のうち長さが最小のものを $W : v_0(= u), e_1, \dots, e_k, v_k(= v)$ とする(歩道が存在し, 長さは非負整数だから最小のものが存在する). この W が道であることを示す. もし頂点に重複があり $v_i = v_j$ ($i < j$) となるとすれば, W から $e_{i+1}, v_{i+1}, \dots, e_j, v_j$ の部分を取り除いた列

$$v_0, e_1, \dots, e_i, v_i(= v_j), e_{j+1}, v_{j+1}, \dots, e_k, v_k$$

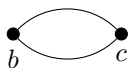
もまた u から v への歩道であり, その長さは W より短い. これは W の最小性に反する. ゆえに W の頂点はすべて相異なり, W は道である. □

この命題により, 第 1 回で定めた連結性(任意の 2 頂点が歩道で結ばれること)は, 「任意の 2 頂点が道で結ばれること」と言い換えてもよい.

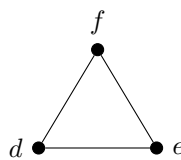
例 2-1 次の図は, 左から長さ 1, 2, 3 の閉路の例である. すなわち a におけるループ(長さ 1), b, c を結ぶ多重辺(長さ 2), 三角形 d, e, f (長さ 3) である.



長さ 1



長さ 2



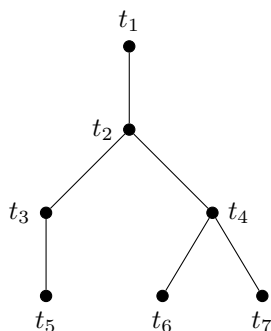
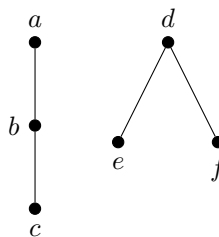
長さ 3

● 2-2 : 木と森

定義 2.4. 連結かつ非巡回的なグラフを **木** (tree) という. また, 非巡回的なグラフ (連結とは限らない) を **森** (forest) という. 次数が 1 の頂点を **葉** (leaf) という.

前項の注意より, 木や森は単純グラフである. また, 森の各連結成分は連結かつ非巡回的, すなわち木である. つまり, 森とはいくつかの木を寄せ集めたものに他ならない.

例 2-2 下図左の T は木であり, 頂点 7 個・辺 6 本で, 葉 (次数 1 の頂点) は t_1, t_5, t_6, t_7 の 4 つである. 下図右の F は森であり, 2 つの連結成分 (木) からなる.

 T (木) F (森)

命題 2.5. 2 個以上の頂点をもつ有限な木は, 少なくとも 2 個の葉をもつ.

証明. T を 2 個以上の頂点をもつ有限な木とする. T の道のうち長さが最大のものを一つとり, $P : v_0, e_1, v_1, \dots, e_k, v_k$ とする (頂点数が有限なので道の長さには上限があり, 最大の道が存在する). T は連結で頂点が 2 個以上だから辺をもち, $k \geq 1$ である.

端点 v_0 が葉であること, すなわち $\deg(v_0) = 1$ を示す. v_0 は e_1 の端点だから $\deg(v_0) \geq 1$ である. もし $\deg(v_0) \geq 2$ ならば, e_1 とは異なる, v_0 に接続する辺 f が存在する. f のもう一方の端点を w とおく (T は単純なので $w \neq v_0$). ここで w が道 P 上にないとすると, $w, f, v_0, e_1, v_1, \dots, e_k, v_k$ は P より長い道となり, P の最大性に反する. 一方 w が P 上にあるなら $w = v_i$ と書けるが, $f \neq e_1$ かつ T が単純 (v_0, v_1 を結ぶ辺は高々 1 本) であることから $w \neq v_1$, したがって $i \geq 2$ である. このとき $v_0, e_1, v_1, \dots, e_i, v_i, f, v_0$ は長さ $i+1 (\geq 3)$ の閉路となり, T が非巡回的であることに反する. いずれも矛盾だから $\deg(v_0) = 1$, すなわち v_0 は葉である.

同様に終点 v_k も葉である. $k \geq 1$ より $v_0 \neq v_k$ だから, T は少なくとも 2 個の葉をもつ. □

定理 2.6. n 個の頂点をもつ木は, ちょうど $n-1$ 本の辺をもつ.

証明. 頂点数 n に関する帰納法で示す. $n=1$ のとき, 木は 1 頂点のみからなり辺をもたないから, 辺数は $0 = 1-1$ で成立する.

$n \geq 2$ とし, 頂点数が $n-1$ の木について主張が成り立つと仮定する. T を n 頂点の木とする. 命題 2.5 より

り T は葉 v をもつ. v に接続する唯一の辺を e とし, T から頂点 v と辺 e を取り除いたグラフを T' とする. T' の頂点数は $n-1$ である. T' が木であることを示そう. まず T' は T の部分グラフだから非巡回的である. 連結性は次のように分かる. T' の任意の 2 頂点 x, y に対し, T の連結性と命題 2.3 より T における x から y への道がとれるが, 葉 v は次数 1 だから道の内部の頂点にはなりえず, この道は v を通らない. よってこれは T' 内の道であり, T' は連結である. 以上より T' は $n-1$ 頂点の木だから, 帰納法の仮定よりその辺数は $n-2$ である. したがって T の辺数は $(n-2) + 1 = n-1$ である. $\square$

命題 2.7. n 個の頂点と c 個の連結成分をもつ森は, ちょうど $n-c$ 本の辺をもつ.

証明. 森 F の連結成分を $F_1, \dots, F_c$ とする. 各 F_i は連結かつ非巡回的, すなわち木である. F_i の頂点数を n_i とおくと, 定理 2.6 より F_i の辺数は n_i-1 である. 相異なる連結成分は頂点を共有せず, また各辺は両端点が属する一つの成分の内部にあるから, F の頂点数は $n = \sum_{i=1}^c n_i$, 辺数は $\sum_{i=1}^c (n_i-1) = \left(\sum_{i=1}^c n_i\right) - c = n-c$ である. $\square$

● 2-3 : 全域木

定義 2.8. グラフ $G = (V, E, \psi)$ の部分グラフ T が, G の全頂点を頂点集合にもつ (すなわち $V(T) = V$) 木であるとき, T を G の **全域木** (spanning tree) という.

定理 2.9. 有限な連結グラフ G は全域木をもつ.

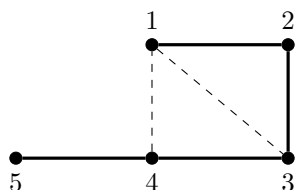
証明. G の全頂点を含む連結な部分グラフ (全域連結部分グラフ) のうち, 辺数が最小のものを T とする (G 自身がそのような部分グラフであり, 辺数は有限なので最小のものが存在する). T が非巡回的であることを示せば, T は全域木である.

T が閉路 C をもつと仮定し, C 上の辺 e (端点を a, b とする) を一つとる. T から e を除いたグラフ $T-e$ もまた G の全域連結部分グラフであることを示す. T の任意の 2 頂点 x, y をとると, T の連結性から x から y への歩道 W がある. W が e を通らなければ, W はそのまま $T-e$ 内の歩道である. W が e を通る場合は, その箇所を, 閉路 C から辺 e を除いて得られる a から b への道で置き換えれば, e を通らない $x-y$ 歩道が得られる. ゆえに $T-e$ においても x, y は結ばれ, $T-e$ は連結である. 頂点集合は変わらないから $T-e$ は全域連結部分グラフであり, その辺数は T より 1 少ない. これは T の辺数最小性に反する. よって T は非巡回的であり, 全域木である. $\square$

命題 2.10. n 個の頂点をもつ連結グラフ G は, $n-1$ 本以上の辺をもつ. さらに, 辺数がちょうど $n-1$ であることと, G が木であることは同値である.

証明. 定理 2.9 より G は全域木 T をもつ. T は n 頂点の木だから, 定理 2.6 よりその辺数は $n-1$ である. $E(T) \subseteq E(G)$ だから $|E(G)| \geq |E(T)| = n-1$ である. 等号 $|E(G)| = n-1 = |E(T)|$ が成り立つとき, $E(T) \subseteq E(G)$ かつ両者の要素数が等しいから $E(G) = E(T)$, すなわち $G = T$ となり G は木である. 逆に G が木ならば, 定理 2.6 より辺数は $n-1$ である. $\square$

例 2-3 下図のグラフ G (頂点 5 個) において, 太線で示した 4 ($= 5-1$) 本の辺は全域木の一例である. 破線の 2 本の辺は全域木に含まれず, これらは G の閉路に対応している.



● 2-4 : グラフの探索

定理 2.9 で連結グラフが全域木をもつことを示したが、その証明は存在を保証するだけであった。ここでは、グラフの全頂点を組織的に訪問し、それと同時に全域木を構成していく手続きである **探索** (search) を扱う。探索は連結成分の決定や経路の発見など、グラフを扱う多くの場面で基本的な役割を果たす。

探索の枠組み. 始点となる頂点 r を一つ選び、「訪問済み」の頂点の集合 S を $S = \{r\}$ から始める。各段階で、一方の端点が S に属し他方が S に属さない辺 $e = \{v, w\}$ ($v \in S, w \notin S$) を一つ選び、 w を新たに訪問して S に加え、このとき用いた辺 e を **木辺** (tree edge) として記録する。そのような辺が存在しなくなれば終了する。記録された木辺全体のなす部分グラフを、始点 r からの **探索木** (search tree) という。

定理 2.11. 有限な連結グラフ G において、任意の始点 r からの探索は全頂点を訪問し、その探索木は G の全域木である。

証明. 各段階で訪問済み頂点はちょうど 1 つ増え、頂点は有限だから手続きは必ず停止する。停止時の訪問済み集合を S とする。もし $S \neq V$ ならば、 G は連結だから S の頂点と $V \setminus S$ の頂点を結ぶ歩道が存在し、その歩道上には S の頂点から $V \setminus S$ の頂点へ渡る辺 $e = \{v, w\}$ ($v \in S, w \notin S$) が現れる。しかしこの e は手続きをまだ続けられることを意味し、停止に反する。ゆえに $S = V$ となり、全頂点が訪問される。

探索木 T を考える。始点 r 以外の各頂点 w は、訪問される際にちょうど 1 本の木辺によって既訪問の頂点に結ばれるから、木辺の総数は $n - 1$ である。また、各頂点は木辺をたどって r に到達できる (訪問順に関する帰納法による。 w を頂点 $v \in S$ からの木辺で訪問したとき、帰納法の仮定より v は木辺で r に結ばれている)。ゆえに T は全頂点を含む連結なグラフであって辺数は $n - 1$ だから、命題 2.10 より T は木、すなわち G の全域木である。□

この定理は、定理 2.9 (全域木の存在) に、それを実際に構成する手続きという形の別証明を与えている。探索の自由度は「各段階で次にどの辺を選ぶか」にあり、その選び方によって代表的な 2 つの方式が得られる。

探索の実装 (OpenList と ClosedList). 上の枠組みを具体的な手続きとして書き下す。枠組みの「訪問済み集合 S 」を **ClosedList** (調査済みリスト, Closed() と書く) で表し、頂点を ClosedList に移すこと (= 枠組みでいう「訪問」) を、その頂点の **調査** (または展開) という。一方、隣接頂点の調査によって見つかったものの、まだ調査していない頂点 (次に調べる候補) のリストを **OpenList** (未調査リスト, Open() と書く) で表す。頂点 B が頂点 A の調査によって見つかったとき、この親子関係を $B[A]$ と記す。 B が初めて調査されるとき $B[A]$ が、枠組みの木辺にあたる。

始点を r とする探索は、次の手続きで与えられる (step.3 のみが幅優先・深さ優先で異なる)。

step.0: Open(r), Closed() と初期化し, step.2 へ。

step.1: OpenList が空なら終了する。そうでなければ step.2 へ。

step.2: OpenList の先頭の頂点 A を取り出す。 A がすでに ClosedList にあれば捨てて step.1 へ戻る。そうでなければ A を ClosedList の末尾に入れ、 A を展開して隣接頂点の集合 V_A を求め、step.3 へ。

step.3 (幅優先): V_A の各頂点 B について、 B が OpenList にも ClosedList にもなければ、 $B[A]$ を OpenList の **末尾** に加える。その後 step.1 へ戻る。

step.3 (深さ優先): V_A の各頂点 B について、 B が ClosedList になければ、 $B[A]$ を OpenList の **先頭** に加える。その後 step.1 へ戻る。

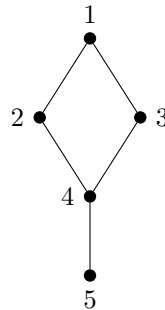
幅優先探索 (breadth-first search, BFS) では OpenList は待ち行列 (先入れ先出し) として働き、始点に近い頂点から距離の層ごとに探索が広がる。**深さ優先探索** (depth-first search, DFS) では OpenList はスタック (後入れ先出し) として働き、行けるところまで深く進んでから後戻り (バックトラック) する。両者は step.3 の挿入位置 (末尾か先頭か) に加えて、重複の判定基準も異なる。幅優先は OpenList と ClosedList の双方を

見るため各頂点は OpenList にただか一度しか入らないが、深さ優先は ClosedList だけを見るため、同じ頂点が一時的に OpenList に重複して現れることがある（その重複は、調査済みの頂点として step.2 で捨てられる）。

いずれの方式でも、各頂点は初めて調査される時、すでに ClosedList にある頂点との辺 $B[A]$ で結ばれる。ゆえに終了時に記録された木辺 $\{B[A]\}$ の全体は始点 r からの探索木をなし、定理 2.11 よりこれは G の全域木である。

幅優先探索の探索木では、 r から各頂点 v への（木辺に沿った）道が、 G における r から v への辺数最小の道（最短路）になることが知られている。ゆえに幅優先探索は、始点からの「辺数で測った距離」を求めるのに用いられる。一方、深さ優先探索では、探索木に含まれない辺はすべて、探索木において一方が他方の先祖となる 2 頂点を結ぶ。この性質は閉路の検出などに応用される。

例 2-4 下図のグラフ G （頂点 $1, \dots, 5$, 辺は $12, 13, 24, 34, 45$ ）を、始点 1 から探索する。展開で得た隣接頂点を OpenList に加えるときは、番号の小さい頂点ほど先に調査されるように、幅優先では小さい順に末尾へ、深さ優先では大きい順に先頭へ加えることにする。



グラフ G

幅優先探索 (BFS) . OpenList の先頭の頂点を調査し、新たに見つかった頂点 $B[A]$ を OpenList の末尾に加える（重複は OpenList・ClosedList の双方で判定）. 次の表に動きを示す（OpenList は左を先頭とする）. 調査の順序は $1, 2, 3, 4, 5$ となる.

取り出す	加える $B[A]$	OpenList (処理後)	ClosedList (処理後)
(初期)	—	1	(空)
1	$2[1], 3[1]$	$2[1], 3[1]$	1
$2[1]$	$4[2]$	$3[1], 4[2]$	1, 2
$3[1]$	(なし)	$4[2]$	1, 2, 3
$4[2]$	$5[4]$	$5[4]$	1, 2, 3, 4
$5[4]$	(なし)	(空)	1, 2, 3, 4, 5

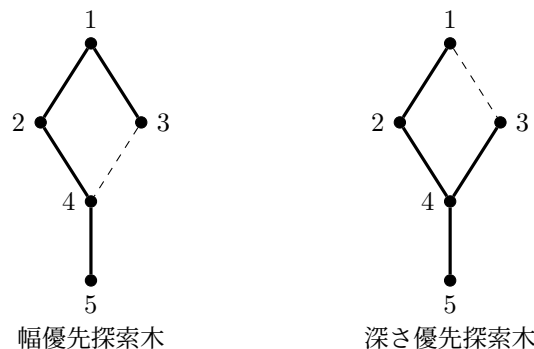
3 の調査では、隣接頂点 4 がすでに OpenList にあるため加えられない（これが非木辺 34 に対応する）. 記録される木辺は $2[1], 3[1], 4[2], 5[4]$, すなわち $12, 13, 24, 45$ である.

深さ優先探索 (DFS) . OpenList の先頭の頂点を調査し、新たに見つかった頂点を OpenList の先頭に加える（重複は ClosedList のみで判定）. 次の表に動きを示す. 調査の順序は $1, 2, 4, 3, 5$ となる.

取り出す	加える $B[A]$	OpenList (処理後)	ClosedList (処理後)
(初期)	—	1	(空)
1	2[1], 3[1]	2[1], 3[1]	1
2[1]	4[2]	4[2], 3[1]	1, 2
4[2]	3[4], 5[4]	3[4], 5[4], 3[1]	1, 2, 4
3[4]	(なし)	5[4], 3[1]	1, 2, 4, 3
5[4]	(なし)	3[1]	1, 2, 4, 3, 5
3[1]	(捨てる)	(空)	1, 2, 4, 3, 5

4 の調査では、隣接頂点 3 がすでに OpenList にあっても、ClosedList のみで判定するため 3[4] として先頭に加わる。このため 3 は親 4 から調査され (木辺 34), 後に残った 3[1] は 3 が調査済みのため捨てられて、辺 13 は非木辺となる。記録される木辺は 2[1], 4[2], 3[4], 5[4], すなわち 12, 24, 34, 45 である。

以上の探索で得られる 2 つの探索木を下図に示す (太線が木辺, 破線が非木辺)。



幅優先探索では、調査は始点からの距離の層 $\{1\}, \{2, 3\}, \{4\}, \{5\}$ (距離 0, 1, 2, 3) の順に進み、探索木における 1 から各頂点への道は G の最短路を与える。一方、深さ優先探索では $1 \rightarrow 2 \rightarrow 4$ と深く潜り、行き止まり (頂点 3, 5) に達すると OpenList (スタック) から順に取り出して後戻りする。二つの探索木が異なるのは、上で見た重複判定の違いに起因する: 頂点 3 の親が、幅優先では 1 (辺 13), 深さ優先では 4 (辺 34) となり、それぞれ辺 34, 13 が非木辺となる。深さ優先探索木の非木辺 13 は、木において 1 が 3 の先祖である 2 頂点を結んでおり、先に述べた性質の一例である。

なお、まだどの探索でも訪問されていない頂点を新たな始点として探索を繰り返せば、グラフのすべての連結成分とその全域木が得られる。これは第 1 回で定義した連結成分を実際に求める方法を与えている。

● 2-5 : 木の特徴づけ

木はいくつかの異なる言い方で特徴づけられる。非単純なグラフは閉路 (ループや多重辺) をもつので木ではない。よって木を論じるには単純グラフの範囲で考えれば十分であり、以下では G を単純グラフとする。

定理 2.12. n 個の頂点をもつ単純グラフ G について、次の 4 条件は同値である。

- (1) G は木である (連結かつ非巡回的)。
- (2) 任意の 2 頂点が、ちょうど 1 本の道で結ばれる。
- (3) G は連結であり、辺数は $n - 1$ である。
- (4) G は非巡回的であり、辺数は $n - 1$ である。

証明. (1) $\Rightarrow$ (2) G は連結だから、任意の 2 頂点 u, v は歩道で、したがって命題 2.3 より道で結ばれる。道が一意であることを示すため、相異なる 2 本の $u-v$ 道 P, Q があったとして矛盾を導く。 P, Q は始点 u を共有

するので, u から両道を同時にたどり, 初めて異なる辺へ分かれる直前の頂点を x とする. P, Q は終点 v も共有するから, x より先で P と Q が再び共有する最初の頂点 y が存在する. このとき P の x - y 間の部分と Q の x - y 間の部分は, 両端 x, y 以外の頂点も辺も共有しないので, 合わせると閉路をなす. これは G が非巡回的であることに反する. ゆえに道は一意である.

(2) $\Rightarrow$ (1) 任意の 2 頂点が道で結ばれるから, 特に歩道で結ばれ G は連結である. もし G が閉路をもてば, G は単純だからその閉路の長さは 3 以上であり, 閉路上の隣り合う 2 頂点は閉路に沿った 2 通りの相異なる道で結ばれる. これは道の一意性に反する. ゆえに G は非巡回的であり, 木である.

(1) $\Rightarrow$ (3) G が木なら連結であり, 定理 2.6 より辺数は $n - 1$ である.

(3) $\Rightarrow$ (1) G を連結で辺数 $n - 1$ とする. 命題 2.10 の後半より, 連結グラフで辺数が $n - 1$ ならば G は木である.

(1) $\Rightarrow$ (4) G が木なら非巡回的であり, 定理 2.6 より辺数は $n - 1$ である.

(4) $\Rightarrow$ (1) G を非巡回的で辺数 $n - 1$ とする. G は森だから, その連結成分数を c とおくと, 命題 2.7 より辺数は $n - c$ である. $n - c = n - 1$ より $c = 1$, すなわち G は連結であり, 非巡回的だから木である. $\square$

最後に, 閉路と木の関係を全体として眺めておこう. n 頂点・ m 辺・ c 連結成分をもつグラフは, 各連結成分の全域木を集めた「全域森」($n - c$ 本の辺をもつ) を含むから $m \geq n - c$ を満たし, 命題 2.7 より等号成立は G が森 (非巡回的) であることと同値である. この非負の量

$$m - n + c$$

を G の 閉路ランク (circuit rank) といい, G を森にするために取り除くべき辺の最小本数を表す. 木とは, $c = 1$ かつ $m = n - 1$, すなわち連結で閉路ランクが 0 のグラフに他ならない.