グラフと組合せ
まとめ

数学的背景

- 集合とその演算
- 数学的帰納法

グラフの記述

- $G=(V,A)$
  - V: 頂点の集合
  - A: 弧の集合
  - 弧からその始点と終点への写像

$$\partial^{+} : A \to V \quad \partial^{-} : A \to V$$

  - 頂点とそこを始点（終点）とする弧

$$\delta^{+} : V \to 2^{A} \quad \delta^{-} : V \to 2^{A}$$

特殊なグラフ

- 完全グラフ (complete graphs)
- 二部グラフ (bipartite graphs)
- 道 (paths)
- 閉路 (circles)
- 木 (trees)
- 極大木 (spanning trees)

二つの探索アルゴリズム

- 深さ優先探索
  - 再帰アルゴリズム
  - 引数は、注目している頂点と、これまでに探索した頂点のリスト
- 幅優先探索
  - 次に調査すべき頂点のキューを使ったアルゴリズム

訂正：幅優先探索

```
L = ∅
Q = {v_0}
while ( Q ≠ ∅ ) {
    v = Q の先頭の要素取り出し
    forall( a ∈ δ⁺v ) {
        w = ∂⁻a
        if ( w ∉ L ){
            Q ← Q ∪ {W}
        }
    }
    L ← L ∪ {V}
}
```

閉路探索

- Euler閉路
  - 一筆描き
  - 全ての弧を一度ずつ経由する閉路
- Hamilton閉路
  - 全ての頂点を一度ずつ経由する閉路
- いずれも再帰的アルゴリズム

ネットワーク

- 弧に重み (実数) が付いている
  - 距離
  - 費用
- 経路や木の重みの最適化問題

最小木

- 弧の重みの和が最小になる極大木 (spanning tree)
- Kruskal アルゴリズム
  - 重み最小の弧を閉路にならないように順に追加
- Jarník-Prim アルゴリズム
  - 始点を定め、木に取り込まれた頂点と残りの頂点を結ぶ重み最小の弧を順に取り込む

最短経路

- 二つの頂点を結ぶ距離 (重み) 最小の有向道
- Dijkstra 法
  - ポテンシャル関数 $p{:}V \to R$ の活用
  - ポテンシャルの確定した頂点の集合W
  - 一度はポテンシャルを計算した頂点の集合U
  - 最短経路に沿った、直前の頂点$q{:}V \to V$

最大フロー

- 弧に容量 $0 \leq c$ と流量 $0 \leq \varphi \leq c$
- Dinitz のアルゴリズム
  - 流量初期化
  - 補助ネットワーク作成
  - 層別ネットワーク作成
  - 極大フローの計算
  - 補助ネットワークを更新する