

計算機ソフトウェア 第七回

電気電子工学科
黒橋禎夫

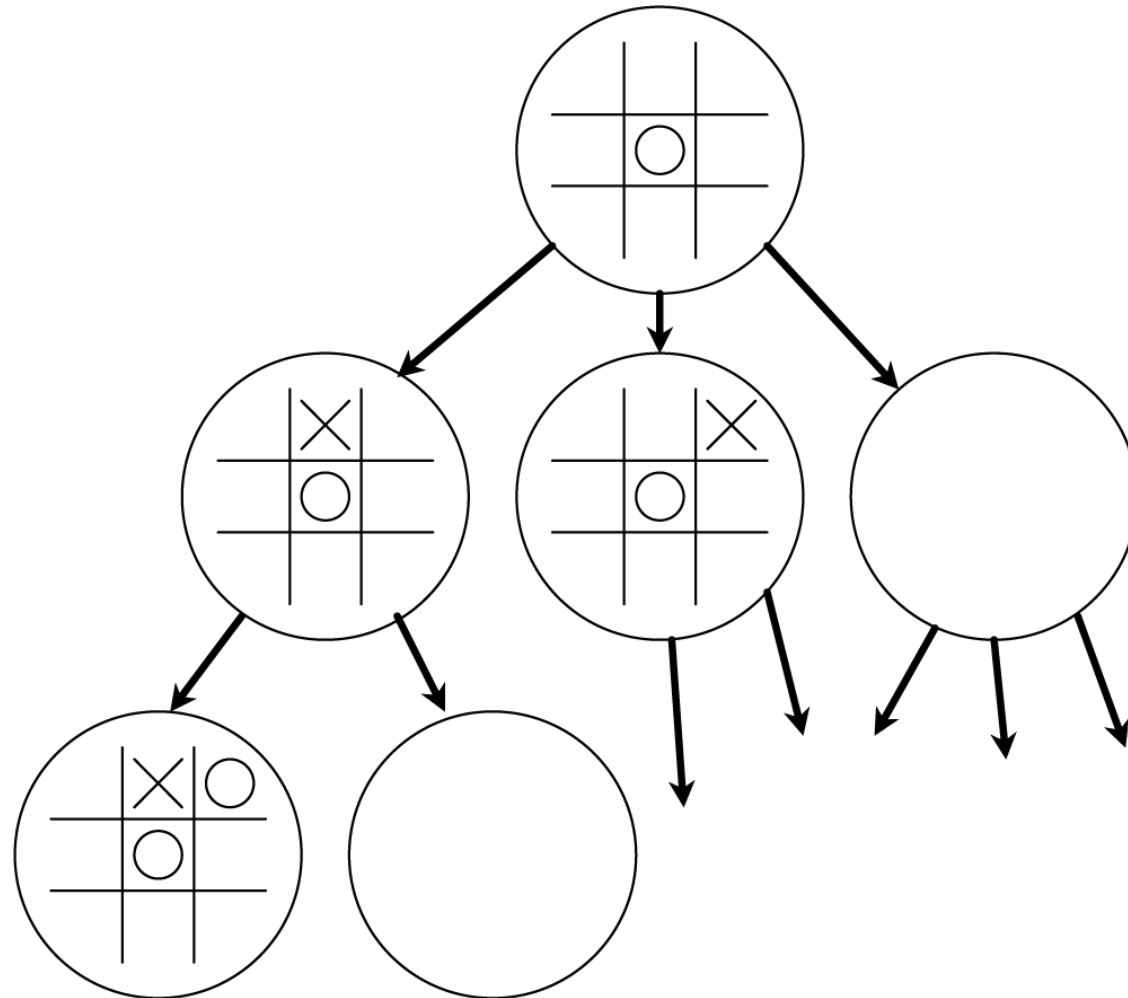
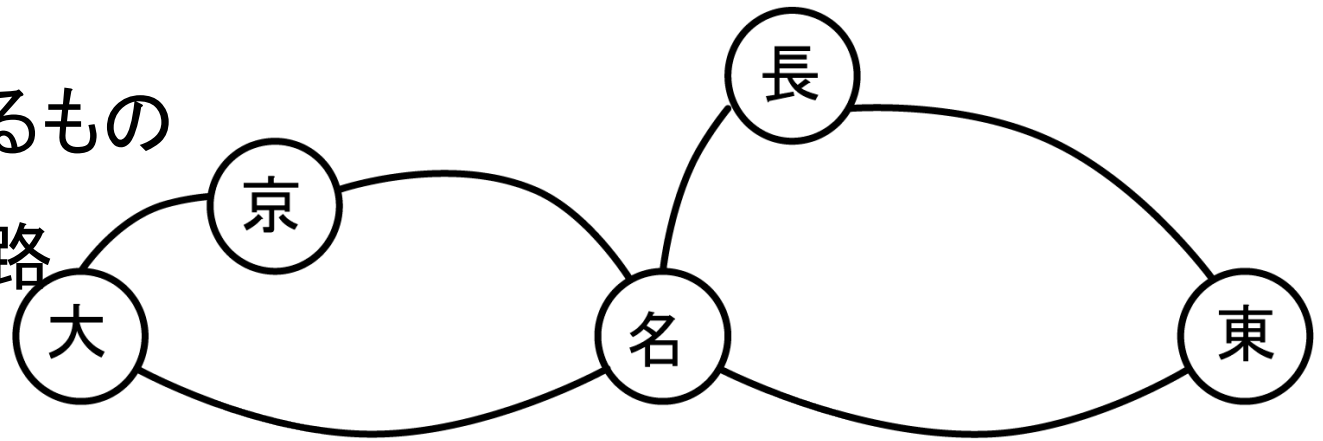
グラフの定義

- グラフは頂点(節点, vertex, node) と 辺(枝, edge, arc, branch) からなる
- 有向グラフ(directed graph): 辺に向きがある
辺(u, v) と 辺(v, u)は違うもの
- 無向グラフ(undirected graph): 辺に向きがない。
辺(u, v) と 辺(v, u)は同じもの

グラフの用語

- 隣接(adjacent): 頂点 v_i と v_j が辺でつながっている
- 道路(path): 隣接する頂点をたどったもの
ex. $v_1 e_1 v_2 e_2 v_3$
- 連結グラフ: 各頂点からすべての頂点へのpathが存在する
- 閉路: 始点と終点が同じ頂点であるpath
- 木(tree): 閉路のない無向連結グラフ

静的に与えられるもの
路線図、電気回路



頂点と辺の作り方が
与えられるもの

ゲームの状態空間
(状態グラフ)

計算機中でのグラフ表現

- 隣接行列

| | 1 | 2 | 3 | 4 |
|---|---|---|---|---|
| 1 | 0 | 1 | 1 | 1 |
| 2 | 1 | 0 | 0 | 0 |
| 3 | 1 | 0 | 0 | 1 |
| 4 | 1 | 0 | 1 | 0 |

- 隣接リスト

1: 2 → 3 → 4

2: 1

3: 1 → 4

4: 1 → 3

行列表現だと0が多く
なって効率悪い

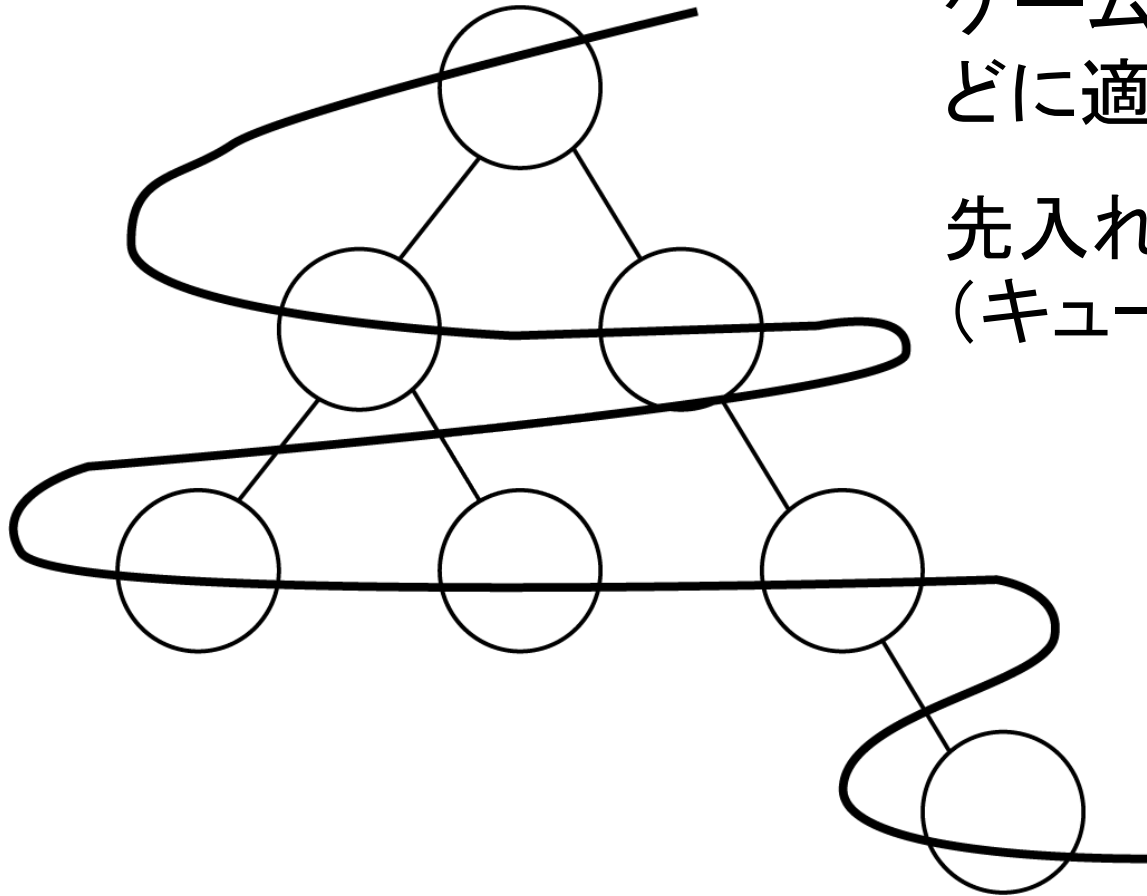
グラフ探索

- グラフのエッジに沿って全てのノードを列挙する課題
- 横型探索
（幅優先探索、breadth-first search）
- 縦型探索
（深さ優先探索、depth-first search）

横型探索

ゲームの状態空間探索などに適用できる

先入れ先出しデータ構造
(キュー、queue)を利用



キュー(queue)

古いものから取り
出すデータ構造

日常生活ではよく
あること

| | | |
|-----|---|--------|
| 1 | 0 | ← head |
| 2 | | ← tail |
| 3 | | |
| ... | | |
| | | |
| | | |

- アルゴリズム GRAPH-SEARCH(横)

入力: G , 初期頂点 v

出力: 頂点のリスト

手続き: 1. 頂点の入れ物 A, B

2. v_0 を A と B に入れる

3. A から一つ頂点 v を取り出し、次を行う

$T(v)$ に属する頂点で B に入っていないものがあれば A と B に入れる

4. A が空になるまで 3. をループ

キューを用いたグラフ探索

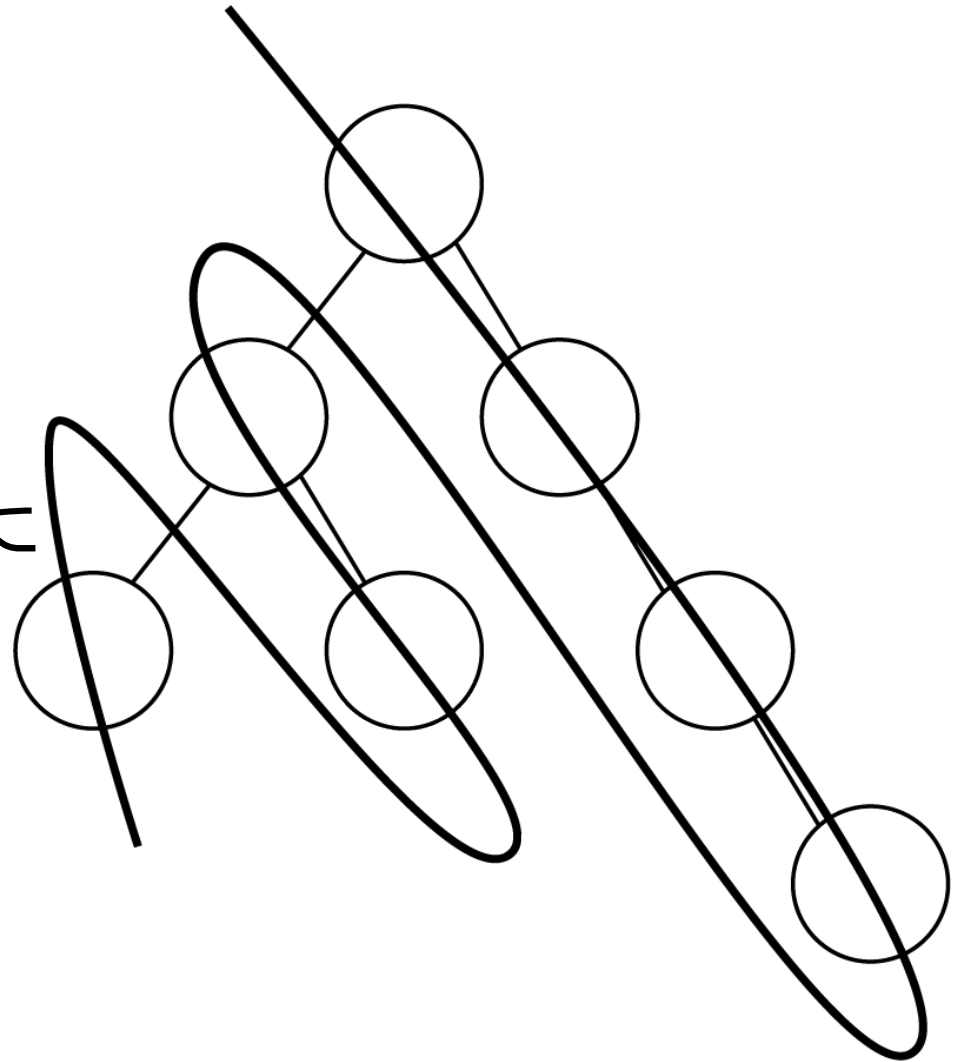
| A | B |
|-----------|--------------------|
| 0 | 0 |
| 1 2 | 0 1 2 |
| 2 3 4 | 0 1 2 3 4 |
| 3 4 5 | 0 1 2 3 4 5 |
| 4 5 | 0 1 2 3 4 5 |
| 5 | 0 1 2 3 4 5 |
| 6 | 0 1 2 3 4 5 6 |
| φ | 0 1 2 3 4 5 6 (出力) |

縦型探索

後入れ先出しデータ構造
(スタック、stack)を利用

再帰による方法と本質的
におなじもの

(\therefore 再帰を実現するためにも
スタックが使われる)

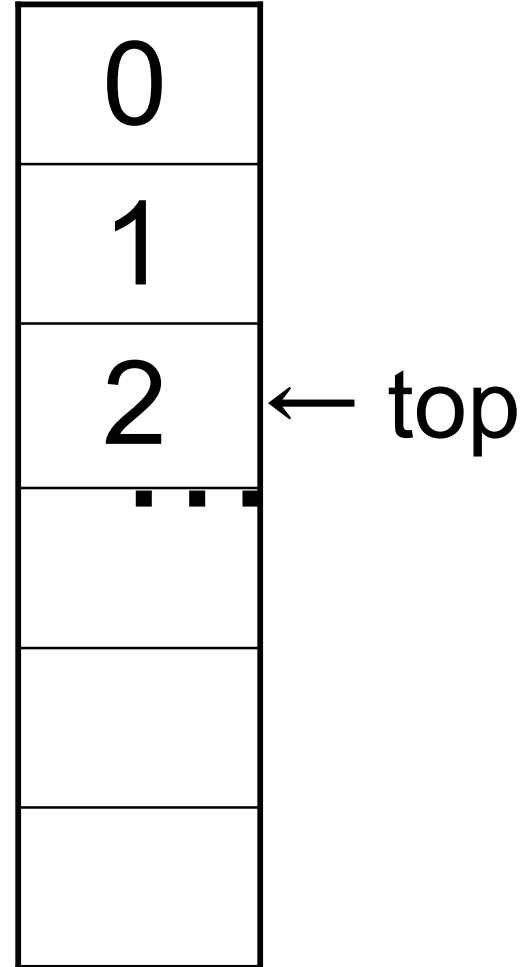


スタック (stack)

新しいものを先に
取り出すデータ構
造

日常生活ではあ
まりみない

計算機の中では
大変よく使われる



- アルゴリズム GRAPH-SEARCH(縦)

入力: G , 初期頂点 v

出力: 頂点のリスト

手続き: 1. 頂点の入れ物 A, B

2. v_0 を A に入れる

3. A から一つ頂点 v を取り出し、次を行う

B に入っていないならば v を B に入れる

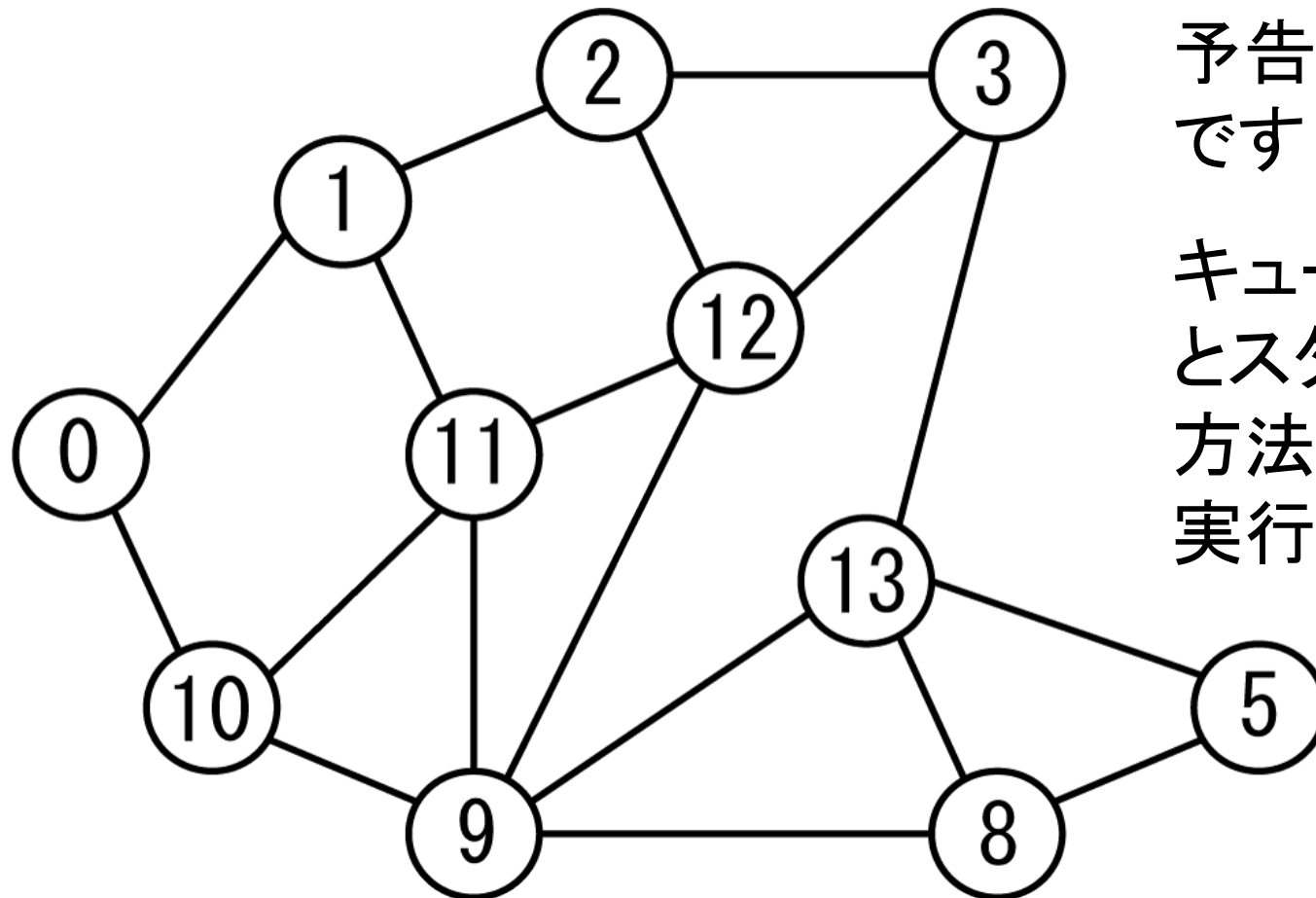
$T(v)$ に属する頂点で B に入っていないものがあれば A に入れる

4. A が空になるまで 3. をループ

スタックを用いたグラフ探索

| A | B |
|---------------|--------------------|
| 0 | \varnothing |
| 1 2 | 0 |
| 1 5 | 0 2 |
| 1 6 | 0 2 5 |
| 1 | 0 2 5 6 |
| 3 4 | 0 2 5 6 1 |
| 3 | 0 2 5 6 1 4 |
| \varnothing | 0 2 5 6 1 4 3 (出力) |

演習 6.5



予告通り路線検索
です

キューを用いる方法
とスタックを用いる
方法で経路探索を
実行しなさい