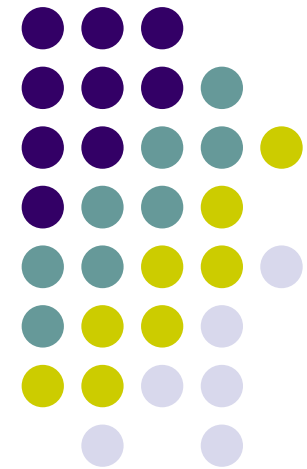


# 情報処理概論 後半7回目



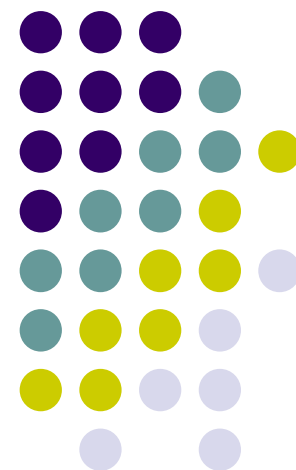
# 今日の内容

- 人工知能



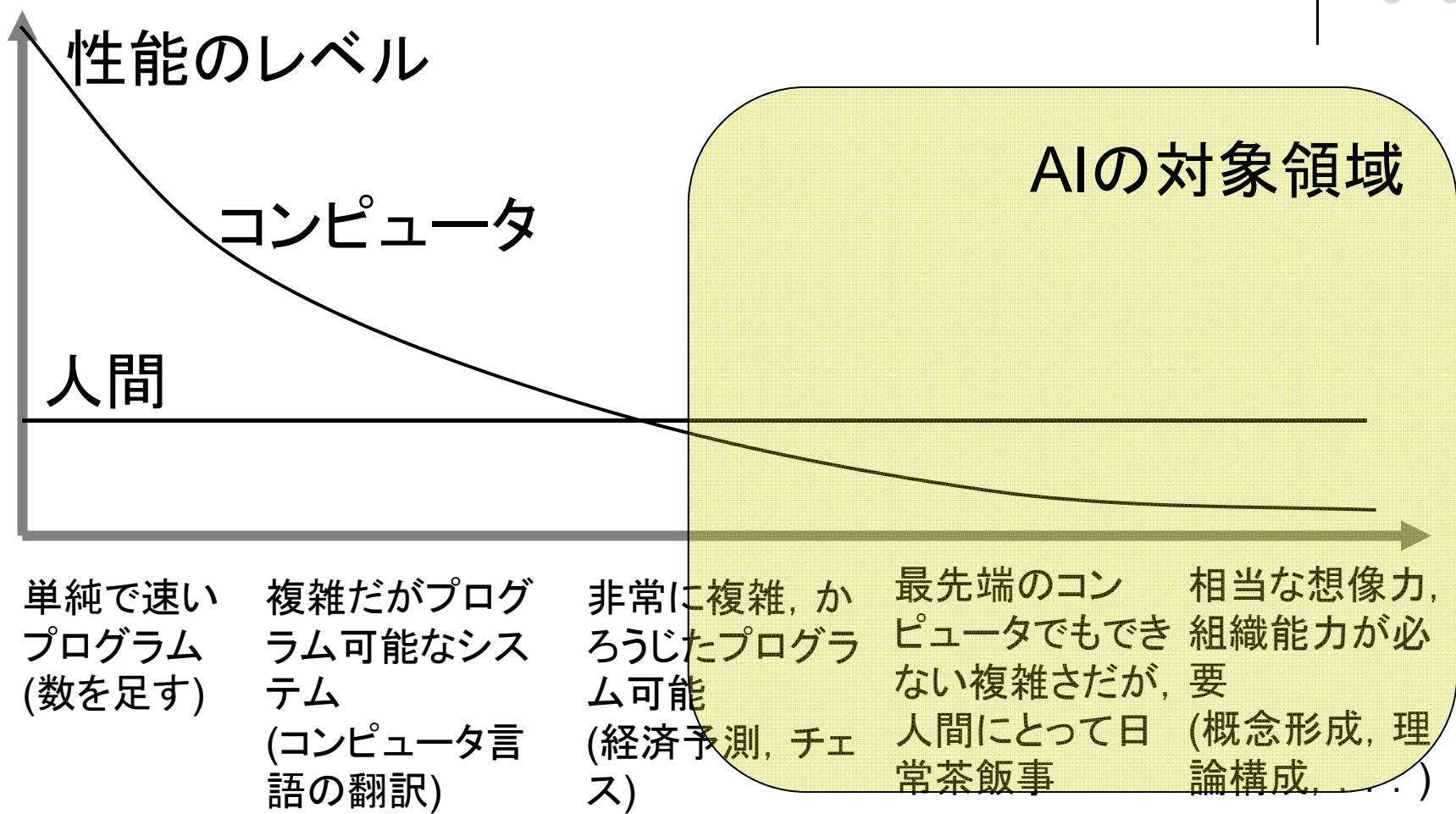
# 人工知能 Artificial Intelligence (AI)

---





# コンピュータの問題解決能力



# コンピュータのプログラムではある のだが...



- 処理の手順が事前に明確になっている  
→ 手続き型プログラミング
- 全体の手順は必ずしも明確ではないが、個々の部品(オブジェクト)の動作と、それらが連携してどう問題を解くかは明確になっている  
→ オブジェクト指向プログラミング

- どのような状態を「問題が解けた」と言うかは明確だが、その問題を解く手順は明確でない

AIが扱う問題

# 人間はそのような問題にどう対処しているか



1. 自分の持っている知識を組合せて解き方を考える
2. 考えた解き方を使って実際に解いてみる
3. 「解けた」状態になれば終了, そうでなければ, 1に戻って違う解き方を考える

試行錯誤

AIとは, このようなことをするプログラム



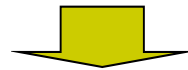
# 知識の表現

- 自然言語による記述
  - 人間が普段使っている言葉で書く
- 形式言語による記述
  - 数学的な言語を設計し、その言語で表現する
- プログラミング言語による記述
  - 既存のプログラミング言語で表現する
- 意味ネットワークによる記述
  - ノードで対象物、名前、特性で表し、アークでノード間の関係を表すグラフで表現する
- 図形による記述
  - 対象物の写真や絵により記述



# 理解

- 椅子の構造に関する知識が記憶されている
- カメラ等で撮ったイメージが記憶されている



- 知識の要素とイメージ中の線や領域をうまく結合できる
  - 各部の接続関係を説明
  - 対象物の役割を識別
  - 知識ベースにある他の詳細の検索



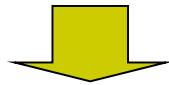
**理解**  
2つの構造間の結合を見いだす



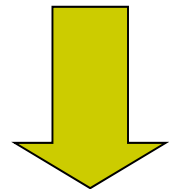


# 学習

- 椅子の概念を持たないコンピュータ



- 椅子のイメージを提示
- 「これは椅子だ」と教える



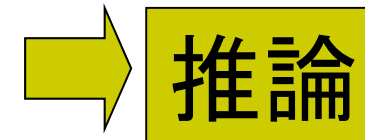
学習

- 椅子がどんなものかわかるようになる
  - 知識構造を形成しイメージと結び付ける



# 推論

- 記憶の中の1つの実体(entity)からもう1つの実体へとつながる結合を発見したり, 構築したりする過程
  - まず, 最初の実体がある
  - 次に, 目標となる実体がある
  - 最初の実体から目標の実体へ向かう道を選択する方法がある
  - 道を選択する方法に基づいて, 最初の実体から目標となる実体への道筋を見つける





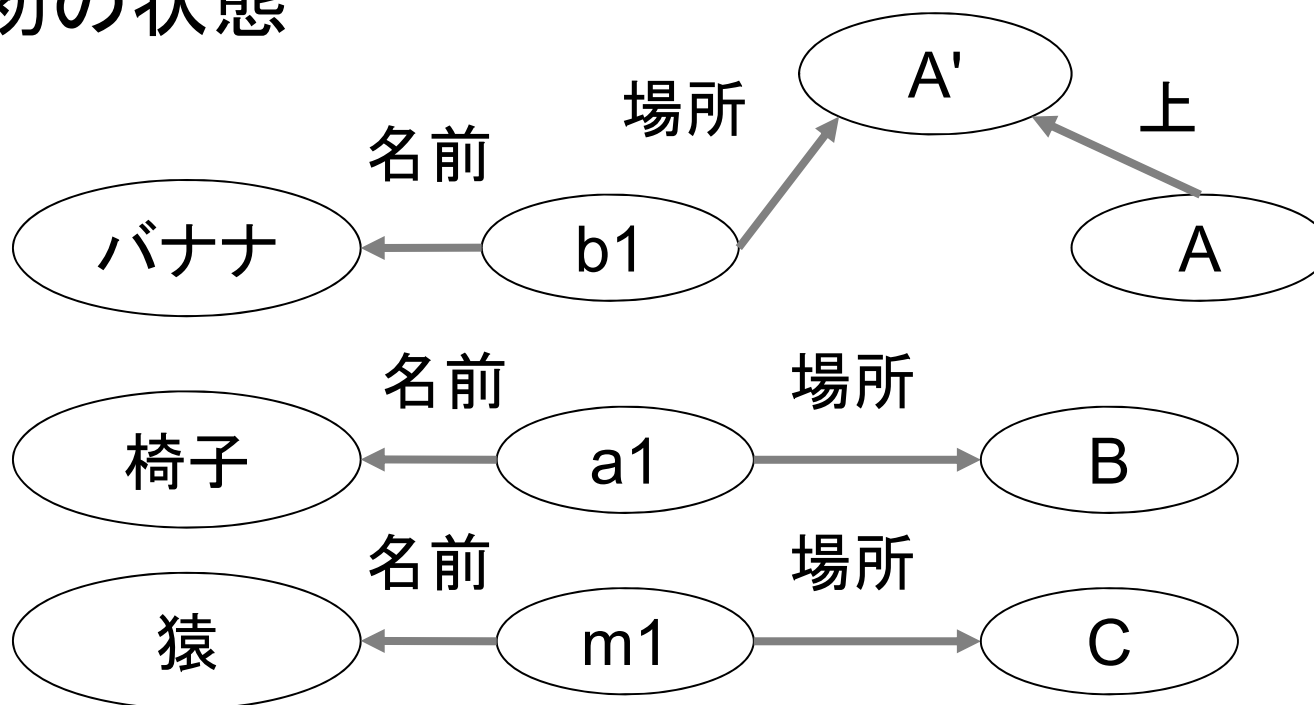
# 推論の例: 猿とバナナ

- 部屋の中に
  - 猿が1匹, 床の上のある点にいる
  - 椅子が1つ, 床の上のある点にある
  - 天井のある点にバナナがぶらさがっている
- 最初の実体: 現在の状態
- 目標の実体: 猿がバナナを手にいれた状態



# 猿とバナナ

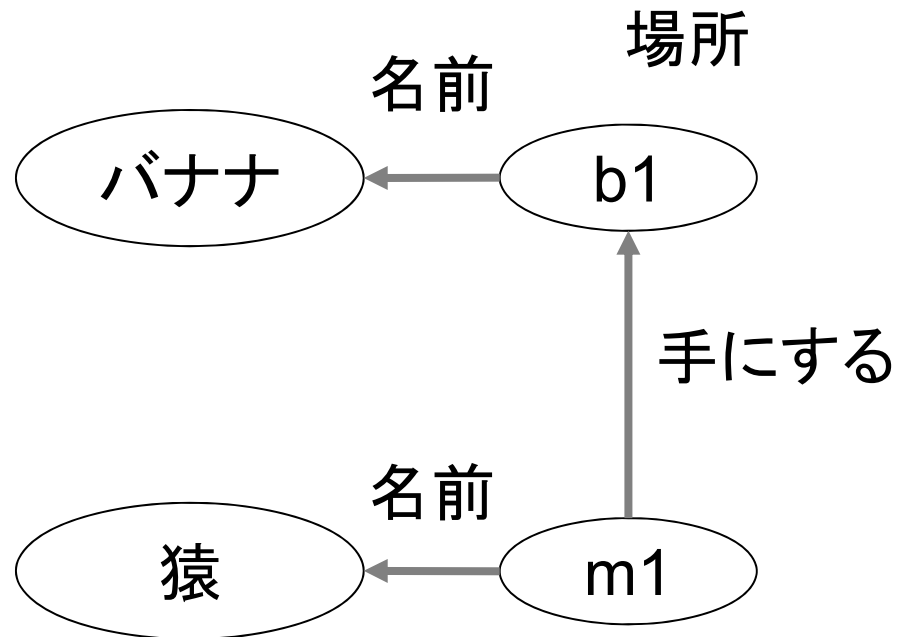
- 最初の状態





# 猿とバナナ (続き)

- 目標の状態





## 推論 (再掲)

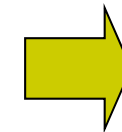
- 記憶の中の1つの実体(entity)からもう1つの実体へとつながる結合を発見したり, 構築したりする過程

- まず, 最初の実体がある

- 次に, 目標となる実体がある

- 最初の実体から目標の実体へ向かう道を選択する方法がある

- 道を選択する方法に基づいて, 最初の実体から目標となる実体への道筋を見つける



推論



# 猿とバナナ (続き)

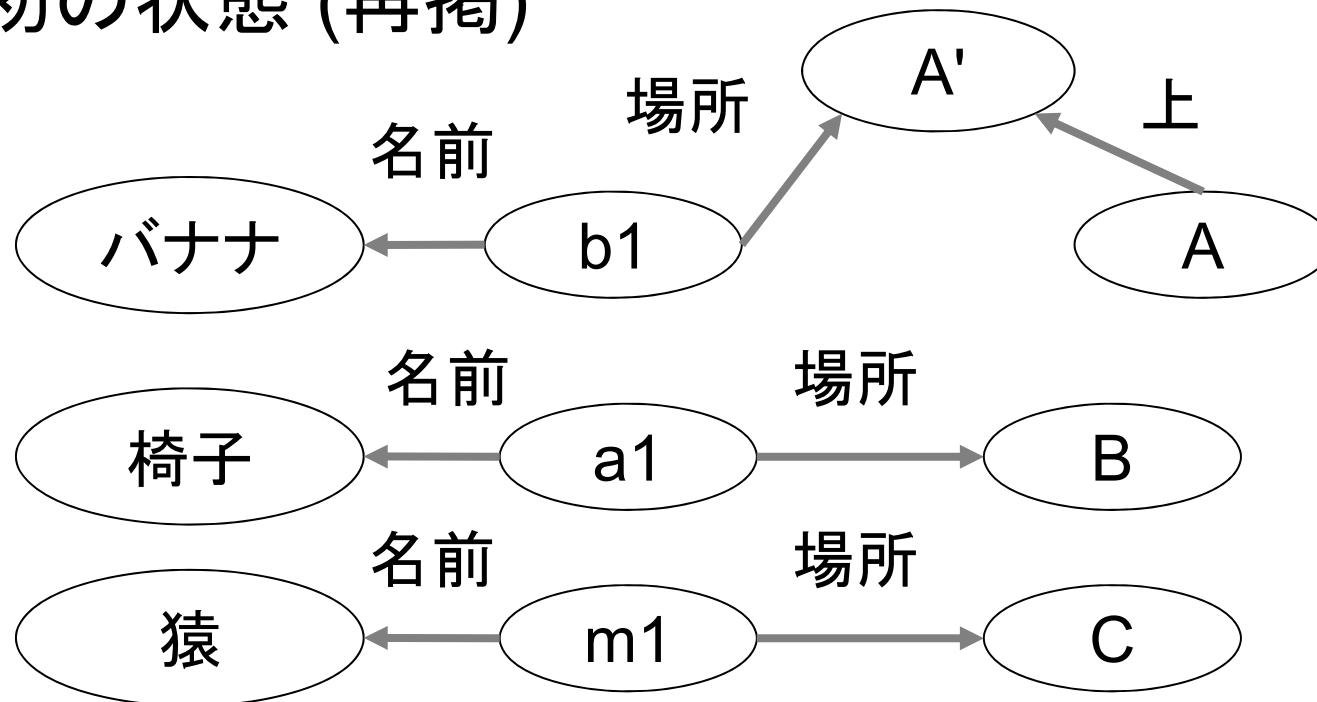
- 動作の定義

| 動作        | 意味         | 条件                     |
|-----------|------------|------------------------|
| go X      | Xへ行く       | 猿は椅子の上にはいないしXにもいない     |
| push X    | 椅子をXへ押して行く | 猿と椅子は同じ位置にあり, 位置はXではない |
| climbup   | 現在の位置で上る   | 猿と椅子は同じ位置にある           |
| grasp     | バナナをつかむ    | 猿とバナナは同じ位置にある          |
| climbdown | 現在の位置で降りる  | 猿は椅子の上にいる              |



# 猿とバナナ (続き)

- 最初の状態 (再掲)

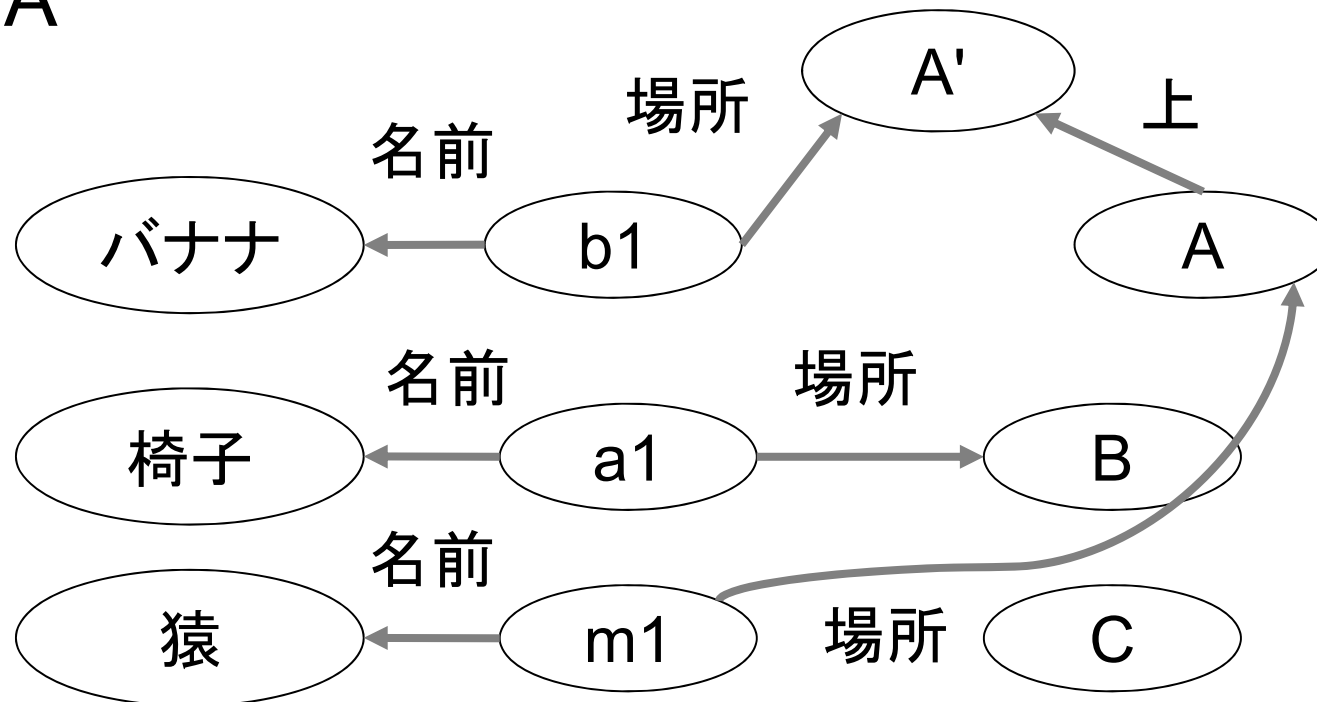






# 猿とバナナ (続き)

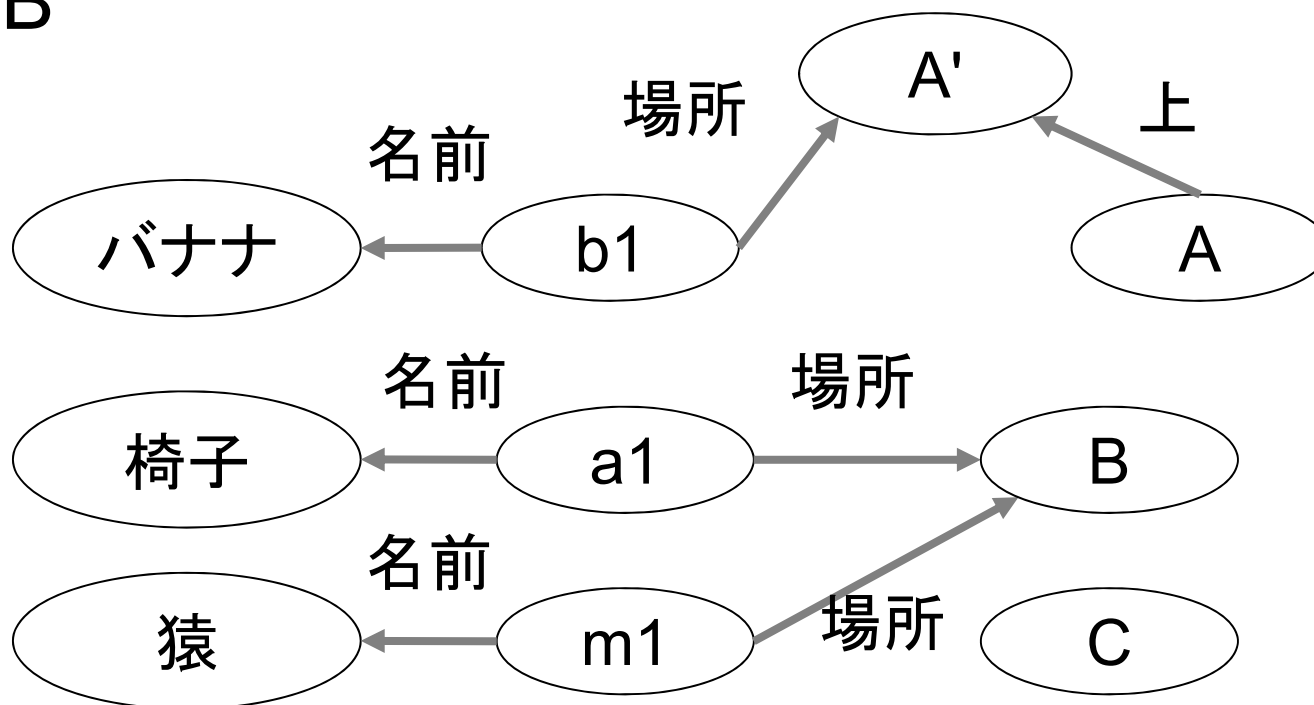
- go A



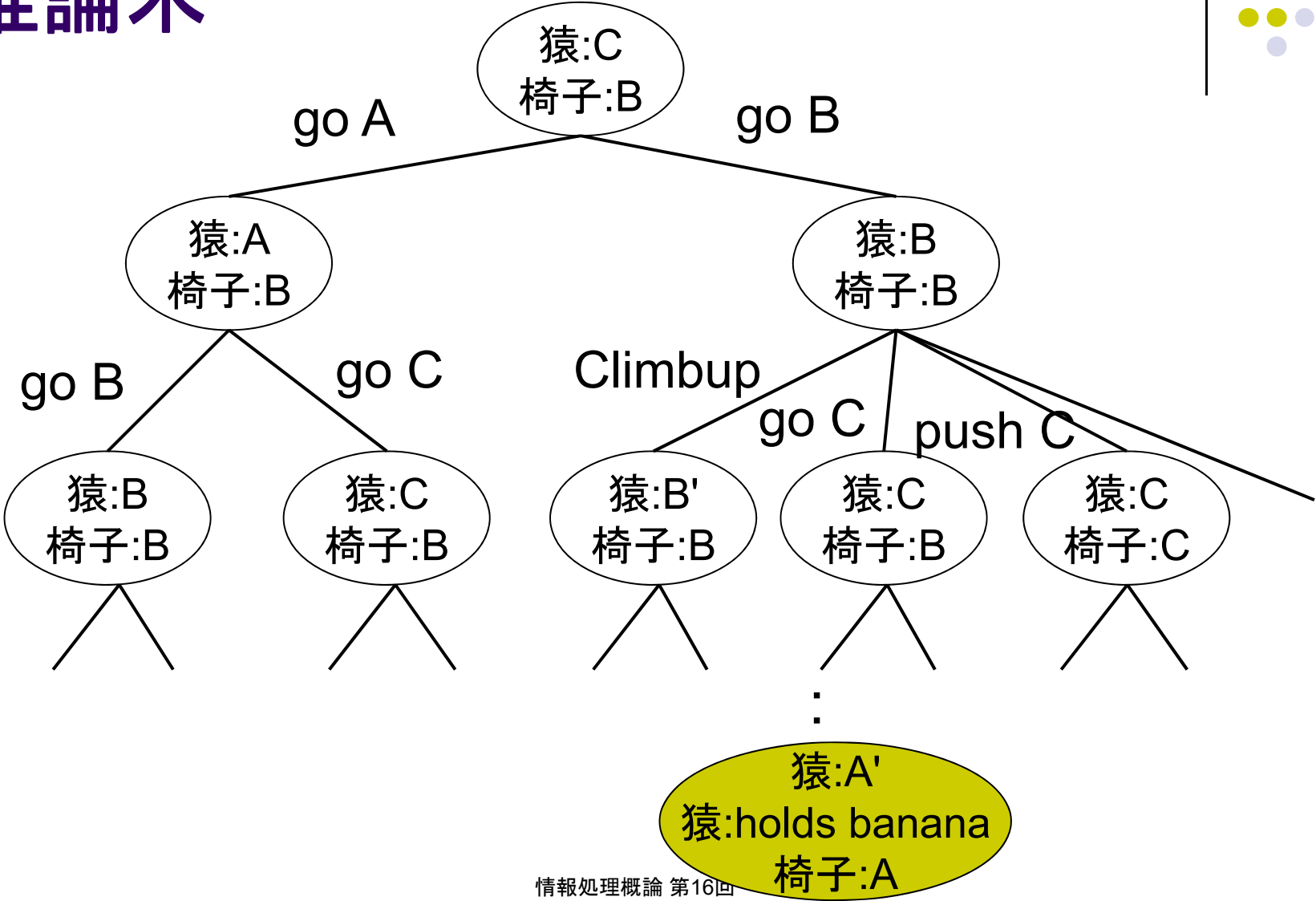


# 猿とバナナ (続き)

- go B



# 推論木

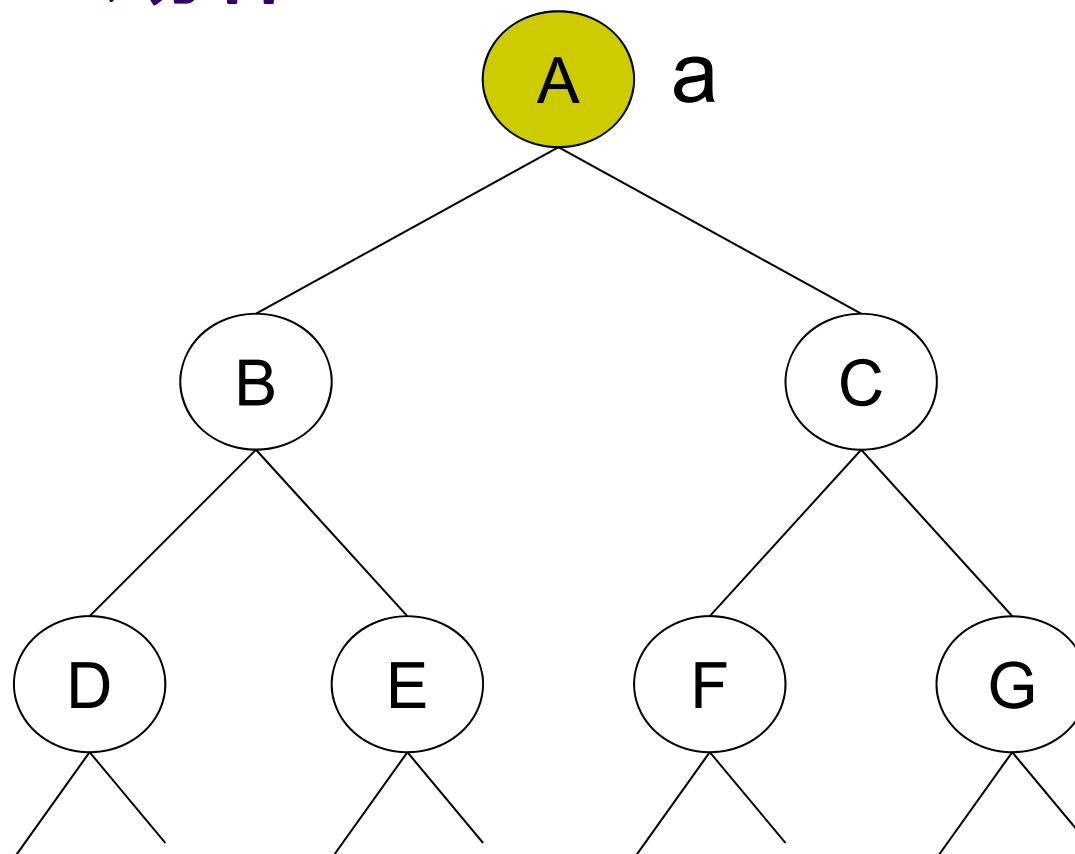




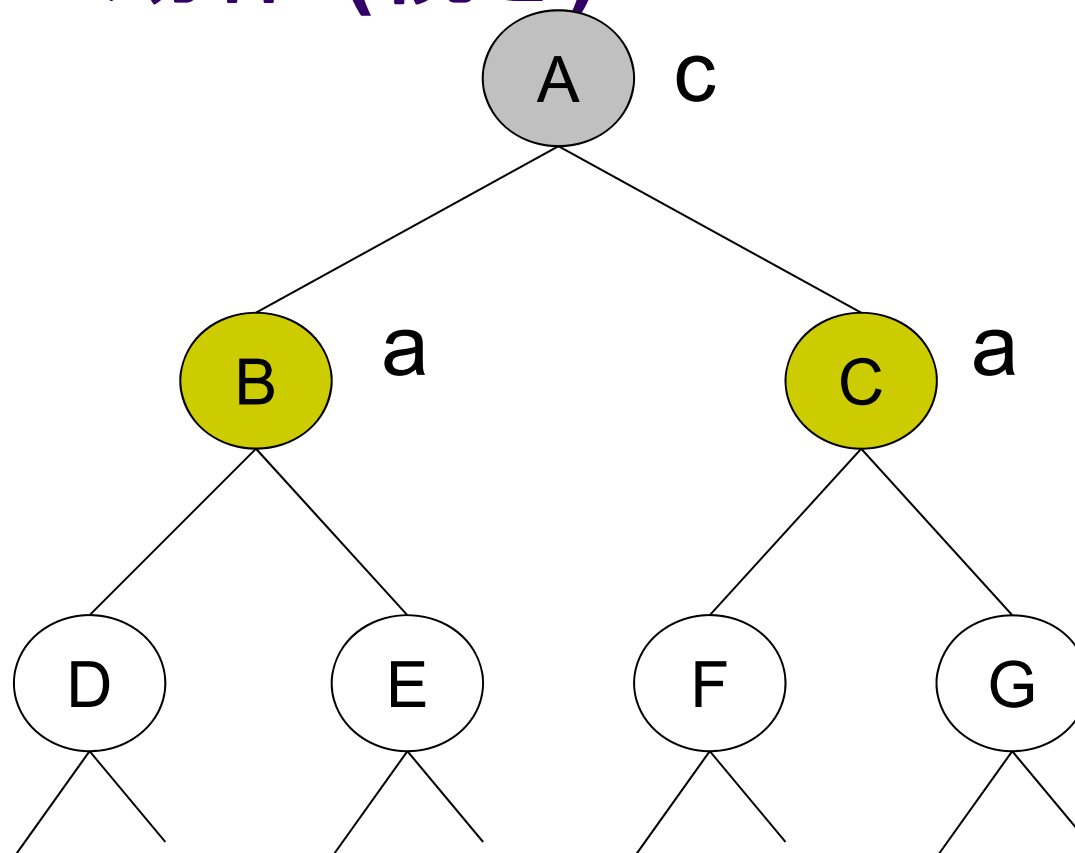
# 推論の動作=探索

- 最初の状態のノードから始める
- 取りうる動作を選び, その動作によって変化した後の状態のノードへ移る(ACTIVEにする)
- 目標状態のノードかどうかを調べる
  - 目標状態のノードなら推論終了(成功)
  - そうでなければ, そのノードをCLOSEにして, そこから他の動作の選択とノードへの移動を続ける
- 目標状態のノードに行き着けるまで続ける

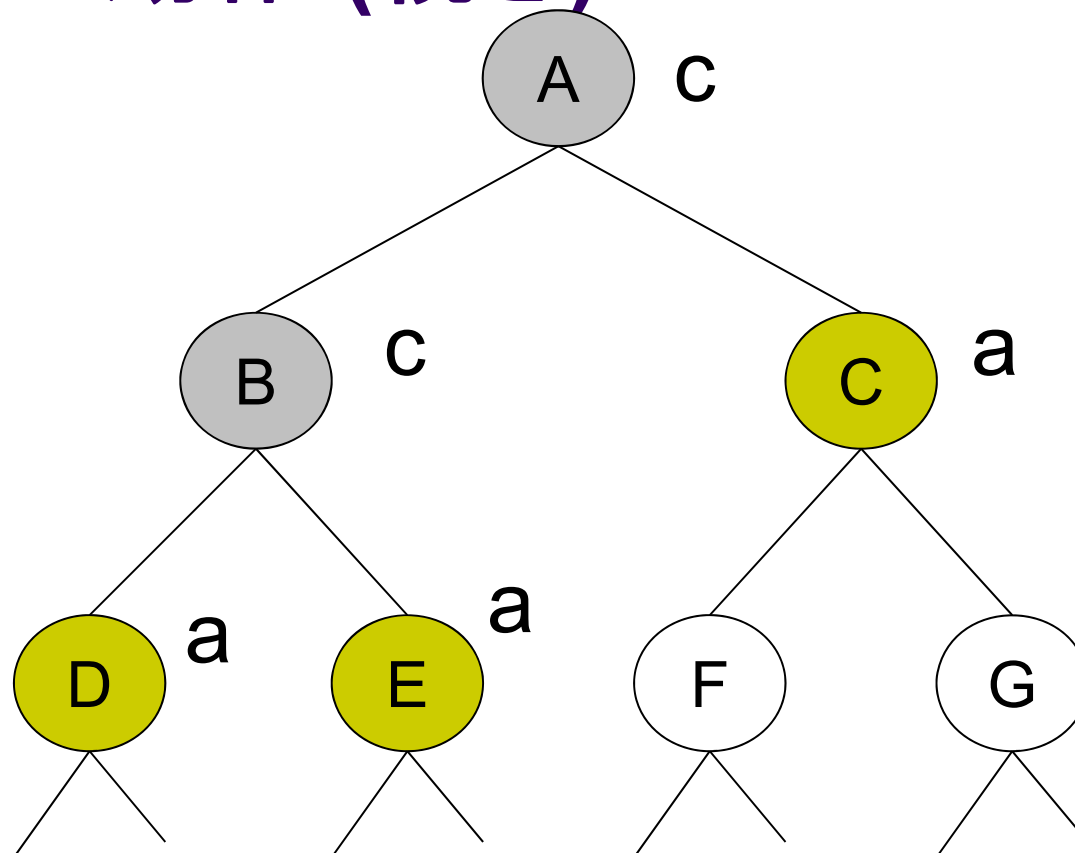
# 推論の動作



# 推論の動作 (続き)



# 推論の動作 (続き)



# 複数のACTIVEノードを どの順に調べるか



- ACTIVEノードの中で、もっとも段数の浅いものを優先する

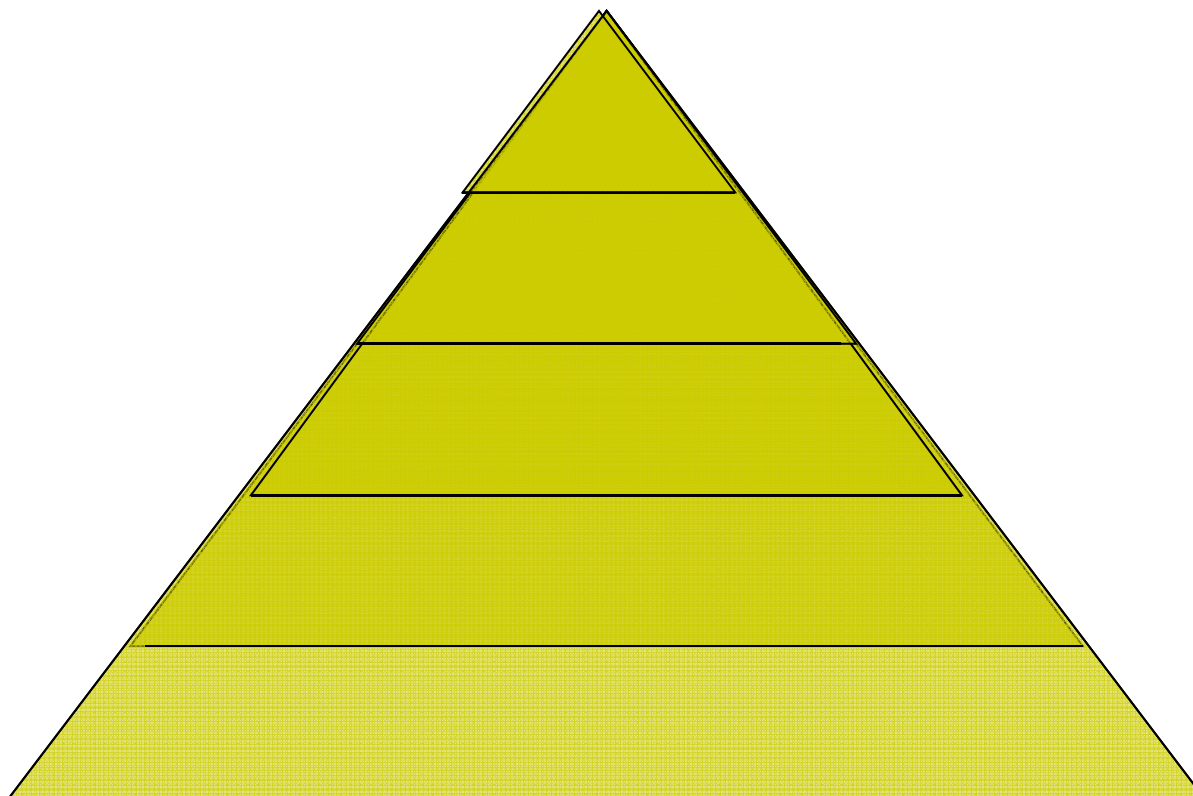
**幅優先探索**

- ACTIVEノードの中で、もっとも左側にあるものを優先する

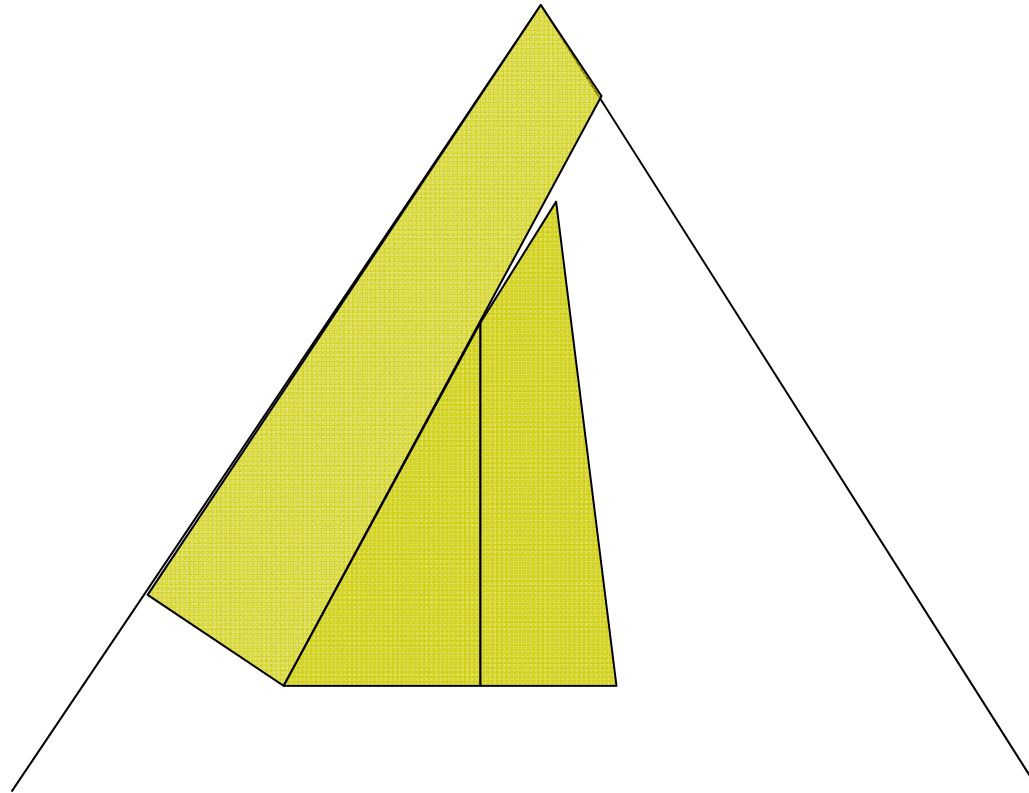
**深さ優先探索**



# 幅優先探索



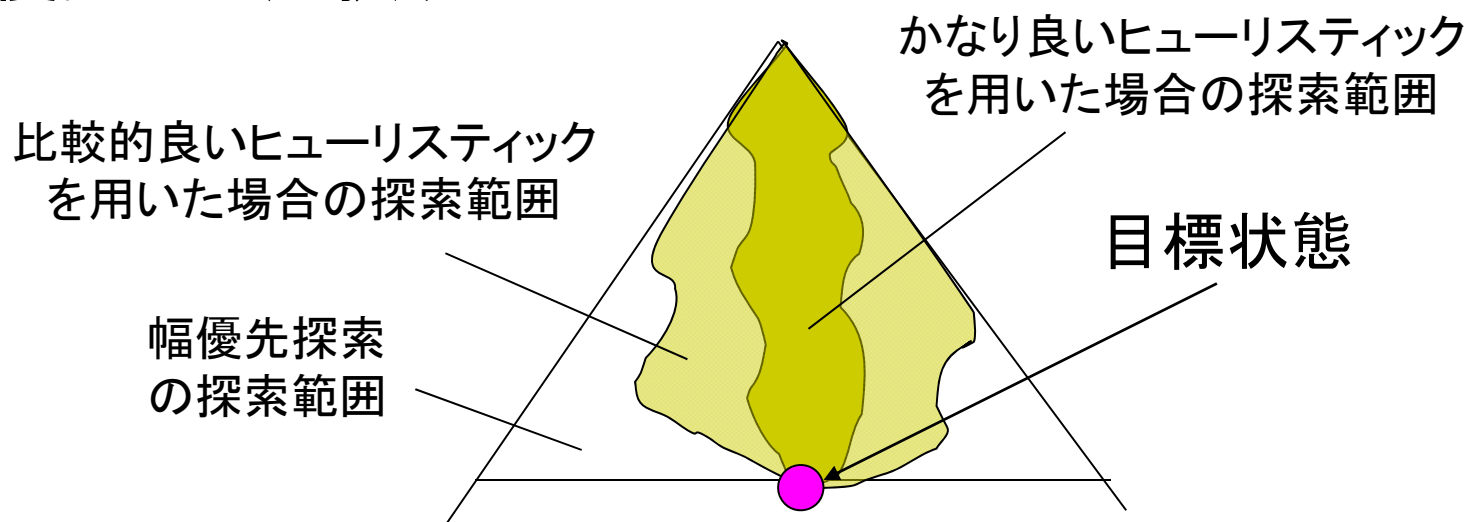
# 深さ優先探索





# ヒューリスティックの利用

- システムが目標の見つけ方について何らかの知識を持っているとする
  - 「ヒューリスティック」と言う
- その知識を用いて、目標状態にいきつけそうなノードを優先的に選択する





# エキスパートシステム

- 人間の専門家のごとく、専門領域において正しい判断を下すコンピュータシステム
- 専門家はif—then—規則を使って知識を表現したがる
  - 「XとYが本当だと認められたなら、Zという状態が起きているとわかる」
  - 「もしZという状態が起きている、AとBは真実なら、私はたいていQを処方する」
- このような規則を一種のコンピュータプログラムのように遂行してくれるアルゴリズムによる推論システムが必要



プロダクション・システム