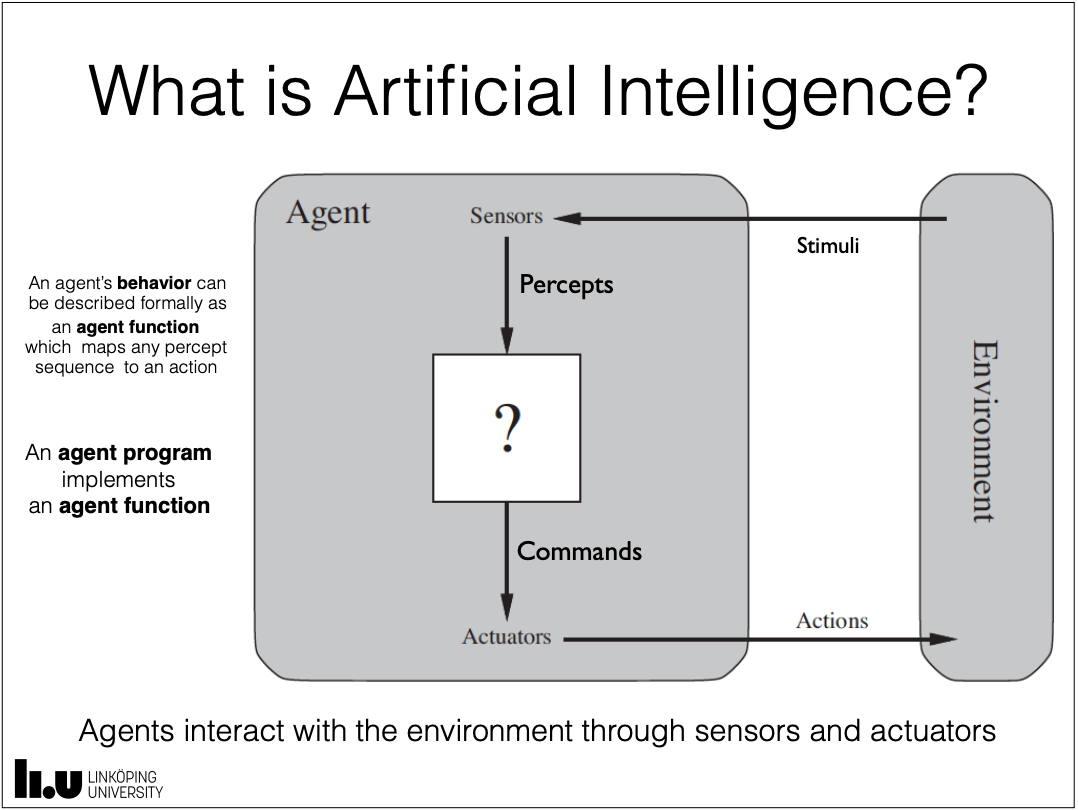
**TDDC17 Artificial Intelligence**

Course content summary

**Seminar 1:**

**What is AI?**

**The Intelligent Agent Paradigm**



**Human-centered AI (Empirical Science, fidelity to human performance):**

Systems that think and act like humans (a human thought process reasoning and behavior)

**Rationality-centered AI (Mathematics/Engineering, ideal concept of Intelligence):**

Systems that think and act rationally (a rational thought process reasoning and behavior)

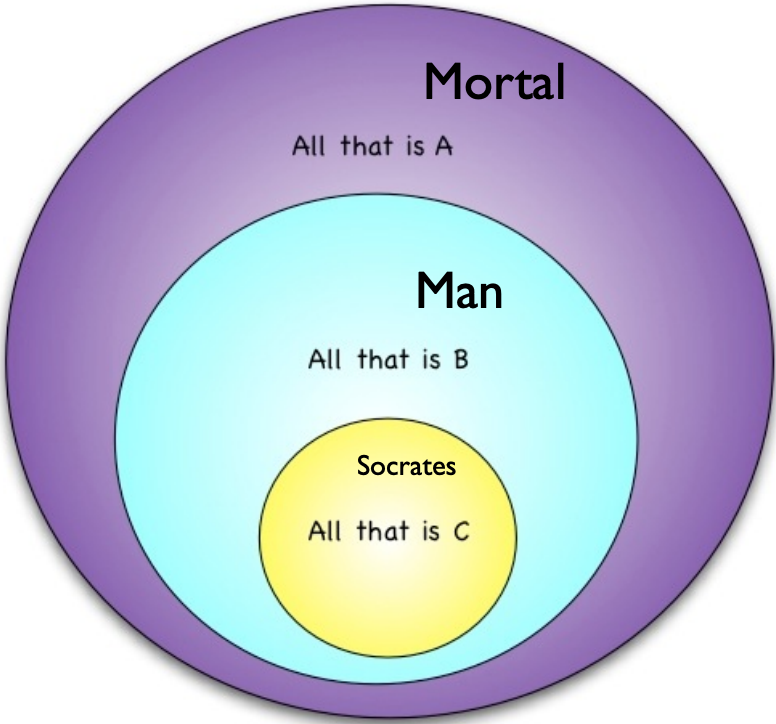
**State of the art achievements in AI research:**

Just AI (Brains without bodies):

* IBM’s Watson: Jeopardy (2011)
* Google Go-Deep Minds: Worlds best Go-player (2015)

Robotics (Bodies without brains):

* Stanford AI lab, Shakey: First robot that could move and see. Able to reason about its own actions. (1966 – 1972)



**Historical Precursors to the Grand Idea of AI:**

**Aristotle (384 – 322 BC):**

(Major premise): All men are mortal.

(Minor premise): Socrates is a man.

------------------------------------------------

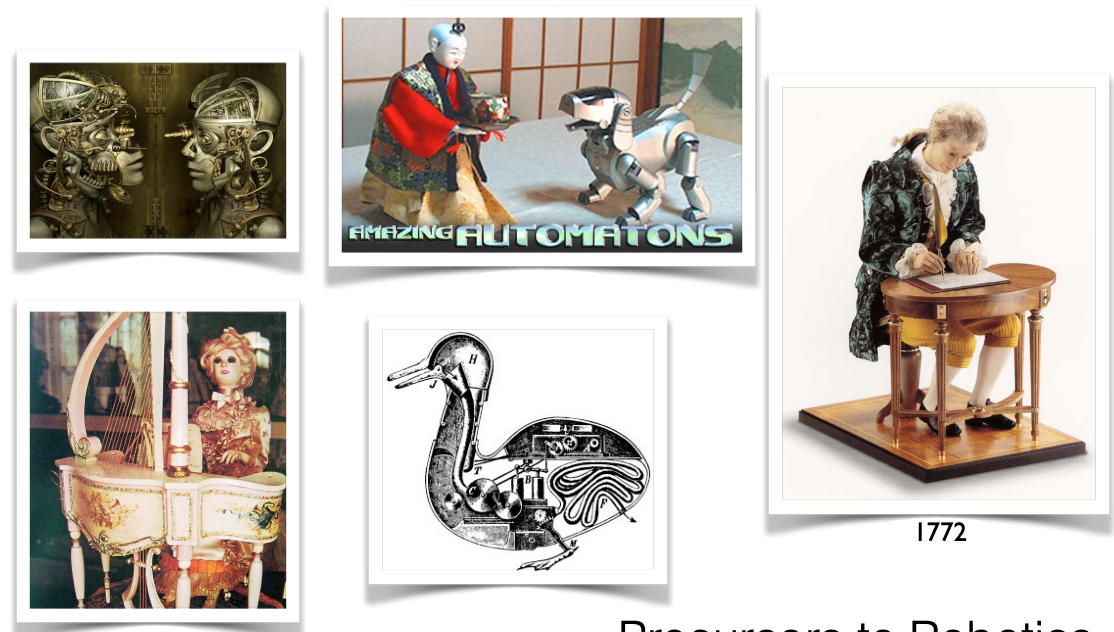
(Deductive Conclusion): Socrates is mortal.

**Leibniz (1646-1716):**

Calculus Rationcinator

* A universal artificial mathematical language
* All human knowledge could be represented in this language
* Calculation rules would reveal all logical relationships among these propositions
* Machines would be able to carry out such calculations

**Automatons (1600 - x):**

****Precursors to robotics

**Boole (1815 – 1864):**

Turned ‘logic’ into algebra.

Classes and terms (thoughts) could be manipulated using algebraic rules resulting in valid inferences.

Logical deduction could be developed as a branch of mathematics.

Subsumed Aristotles syllogisms. In essence Leibniz calculus rationator (lite).

**Frege (1848 – 1925):**

Begriffsschrift “Concept Script”.

1st example of formal artificial language with formal syntax.

Logical inference as purely mechanical operations (rules of inference).

*Intention was to show that all of mathematics could be based on logic! (Logicism)*

Russell’s Paradox:

Frege’s arithmetic made use of sets of sets in the definition of number.

Russell showed that use of sets of sets can lead to contradiction.

Ergo... the entire development of Frege was inconsistent.

**Russel (1872 – 1970):**

Principa Mathematica (Russel & Whitehead)

An attempt to derive all mathematical truths from a well-defined set of axioms and inference rules in symbolic language.

**Hilbert (1862 – 1942):**

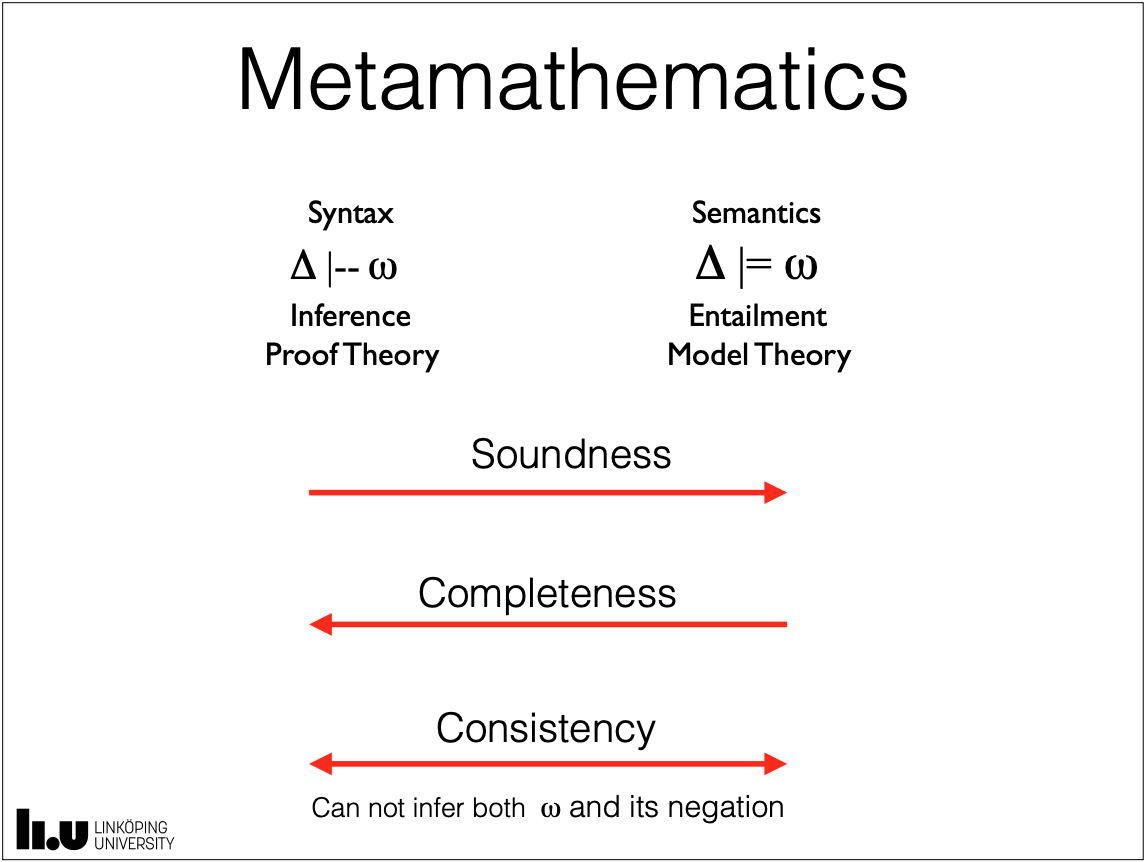
1st problem: Decide the truth of Cantor’s Continum Problem

2nd problem: Establish the consistency of the axioms for the arithmetic of real numbers.

“24 problems for the 20th century”

Hilberts program:

Logic from the outside

Metamathematics

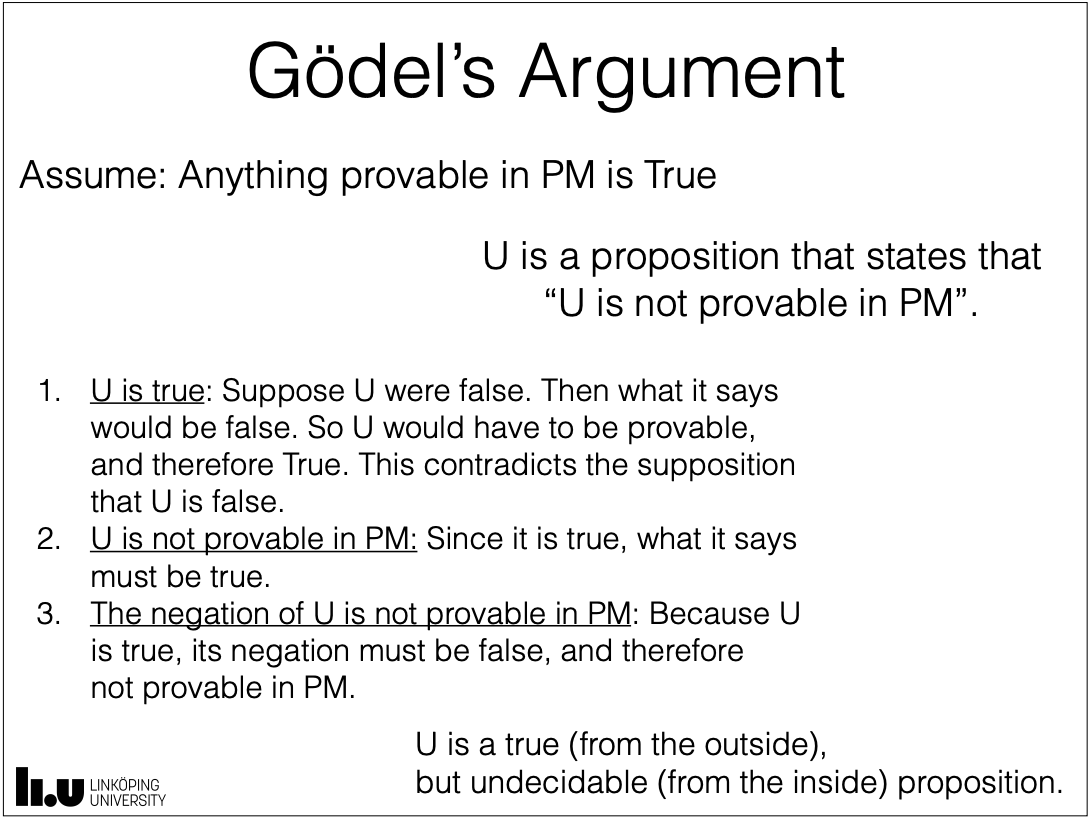
Proof theory

🡪🡪🡪🡪🡪

Logic from the inside

Formal axiomatic theories

Peano Arithmetic

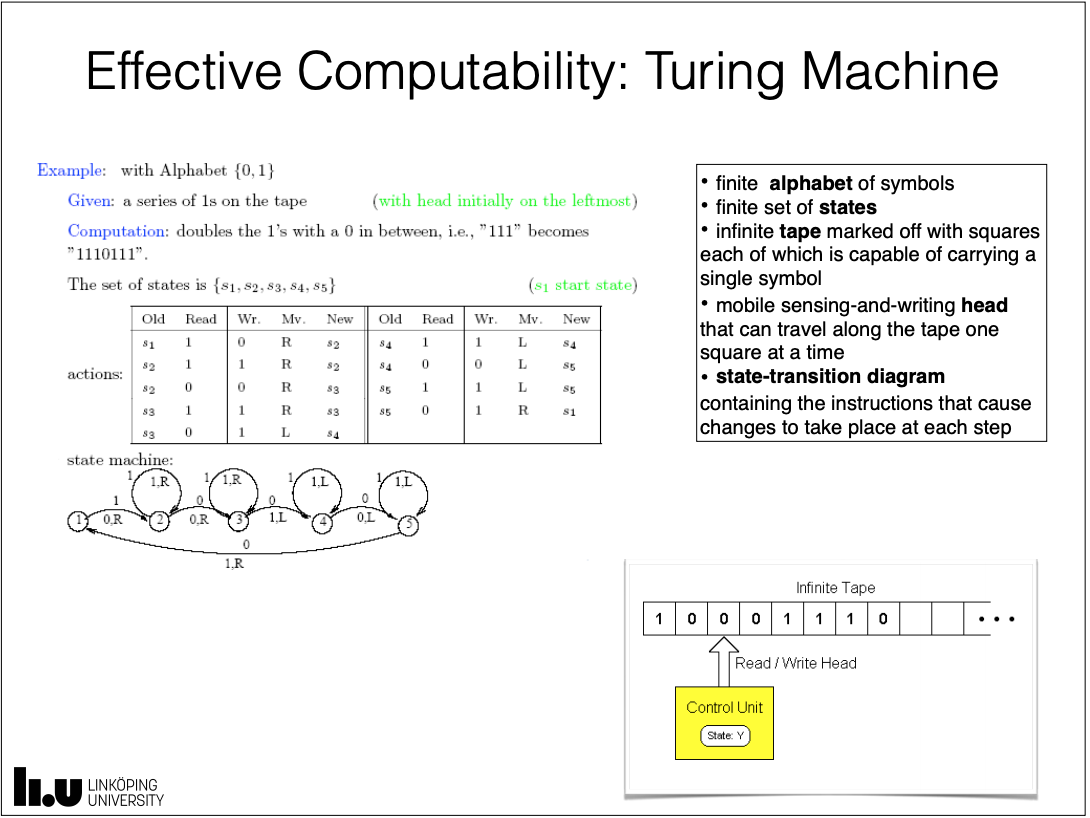


**Gödel (1906 – 1978):**

Showed the completeness of 1st-order logic in his PhD Thesis

Develop metamathematics inside a formal logical system by encoding propositions as numbers.

The logic of PM (and consequently PA) is incomplete. There are true sentences not provable within the logical system.

**Turing (1912 – 1954):**

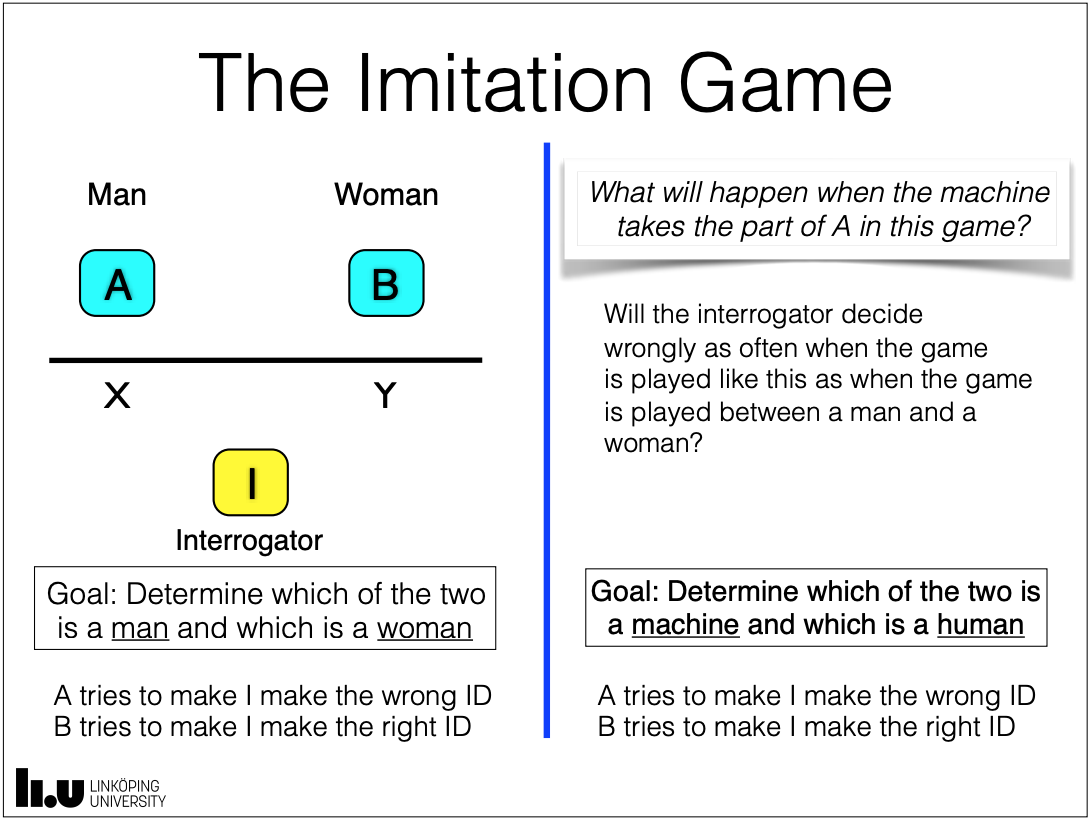
Turing wanted to disprove the 23rd problem (Hilbert).

Found a mathematical model of an all-purpose computing machine.

**The Turing Test:**

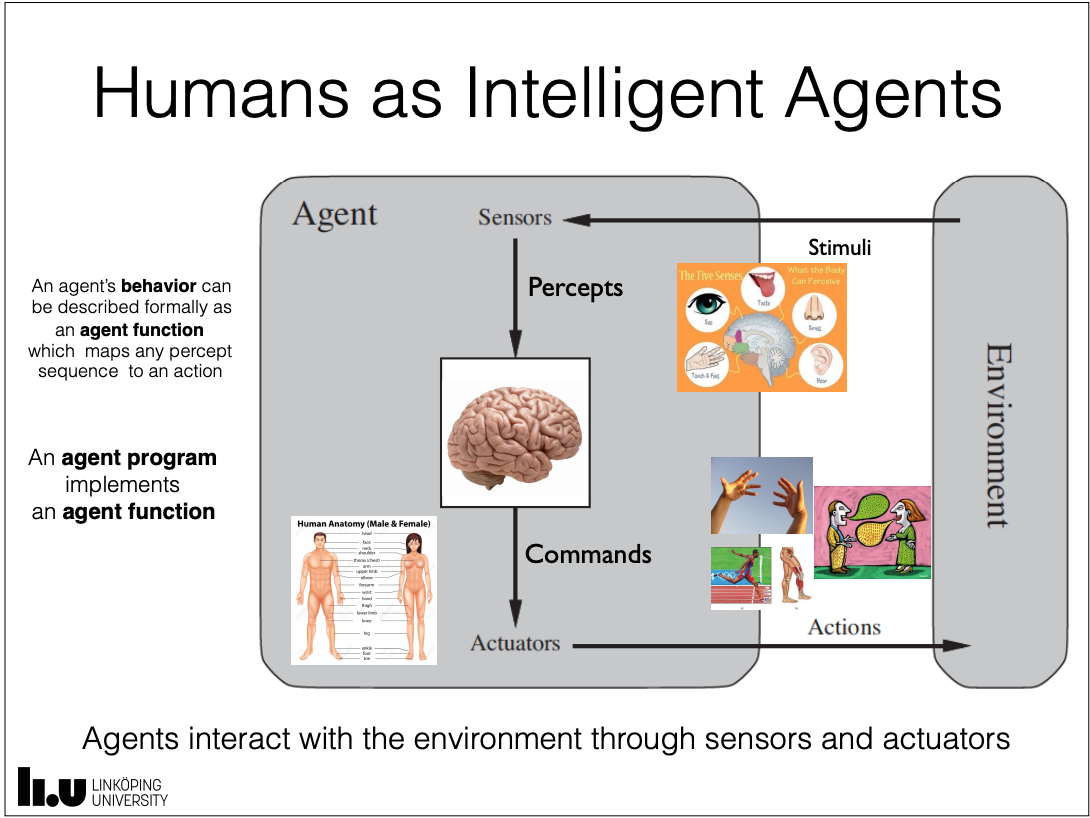
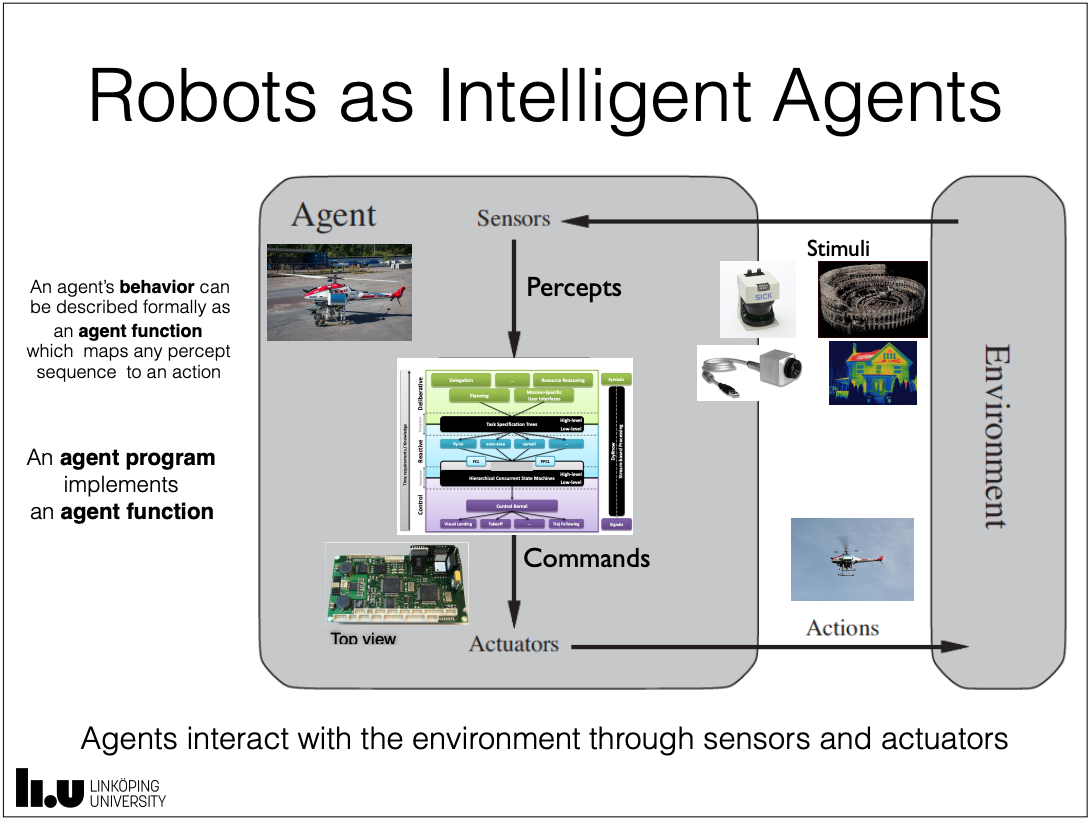
“I propose to consider the question, ‘Can machines think?’”

Meaning of ‘machine’ and ‘think’ is ambiguous, Turing replaces question by another:

“The imitation game”

Can an interrogator decide if A or B is a machine?

**Intelligent Agents:**

“An agent is anything that can be viewed as perceiving its environment through sensors and acting upon that environment through actuators.”

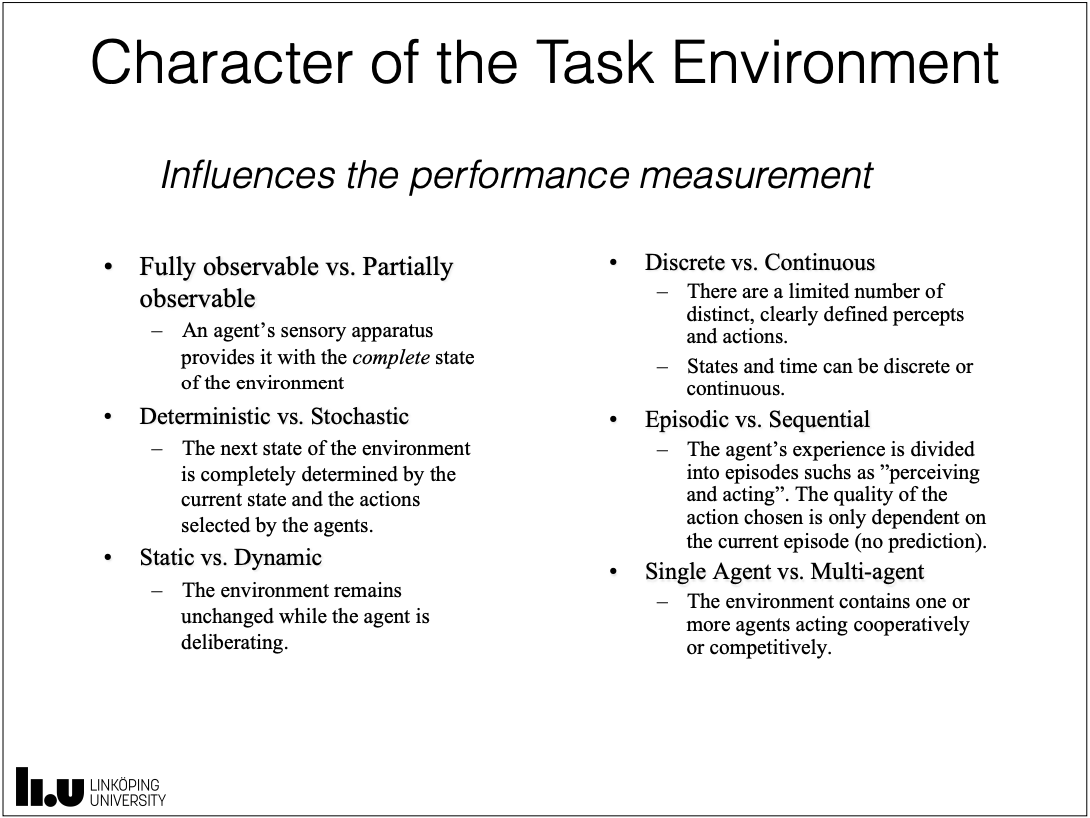
Intelligent agent paradigm:

Evolutionary AI:

Progression of agents (AI systems) each more complex than its predecessor.

Progression loosely follows milestones in evolution of animal species.

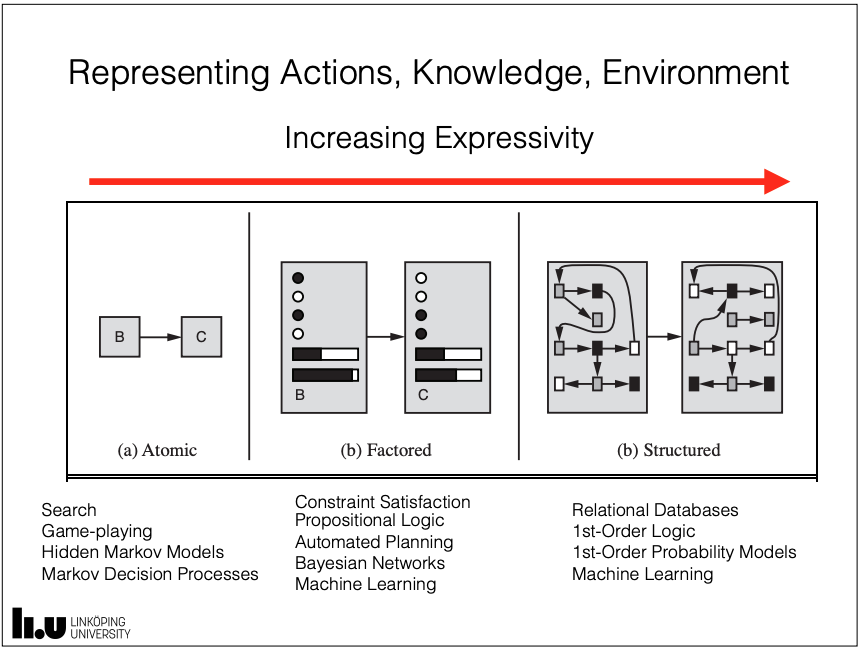
Incrementally introduces techniques for exploiting information about tasks **not** directly sensed.

**Rationality:**

Rationality is dependent on:

* An agents percept sequence; everything the agent has perceived so far.
* The embedding environment; what the agent knows about the environment.
* An agent’s capabilities; the actions the agent can perform.
* The external performance measure used to evaluate the agent’s performance.

An **ideal rational agent** is one that “does the right thing”.

**Agent types:**

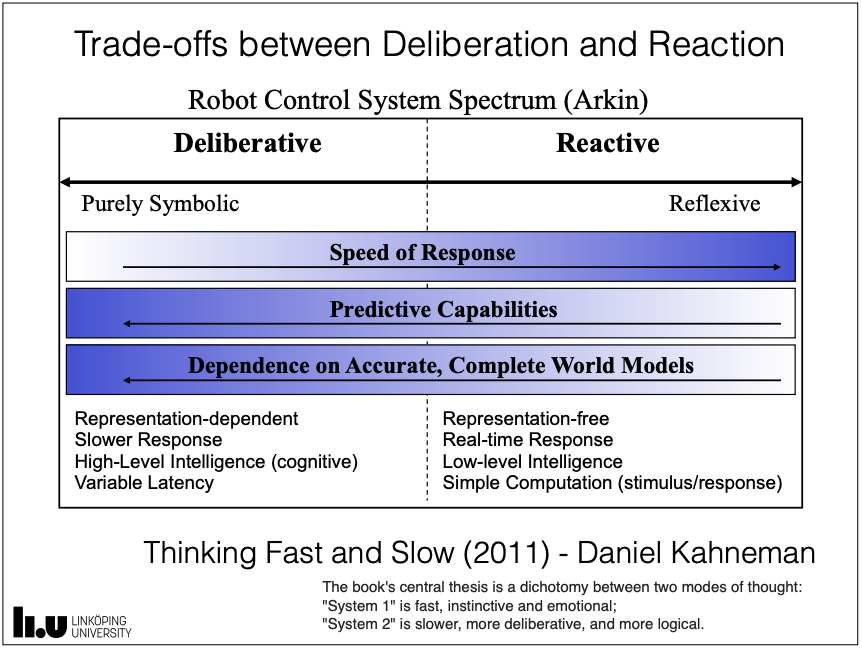
**Simple reflex agent:**

* Stimulus-response agent.
* Reacts to immediate stimuli in the environment.
* No internal state.
* Uses current state of the environment derived from sensory stimuli.

**Model-based reflex agent:**

* Reflex agent with internal state.
* Limited internal state (implies memory).
* Environmental state t+1 is a function of:
  + The sensory input at t+1.
  + The action taken at time t.
  + The previous environmental state at t.

**Goal-based agent (“agents with purpose”):**

* Rich internal state.
* Can anticipate the effects of their actions.
* Take those actions expected to lead toward achievement of goals.
* Capable of reasoning and deducing properties of the world.

**Utility-based agent:**

* Use of utility function that maps state (or state sequences) into real numbers.
* Permits more fine-grained reasoning about what can be achieved, what are the trade-offs, conflicting goals etc.

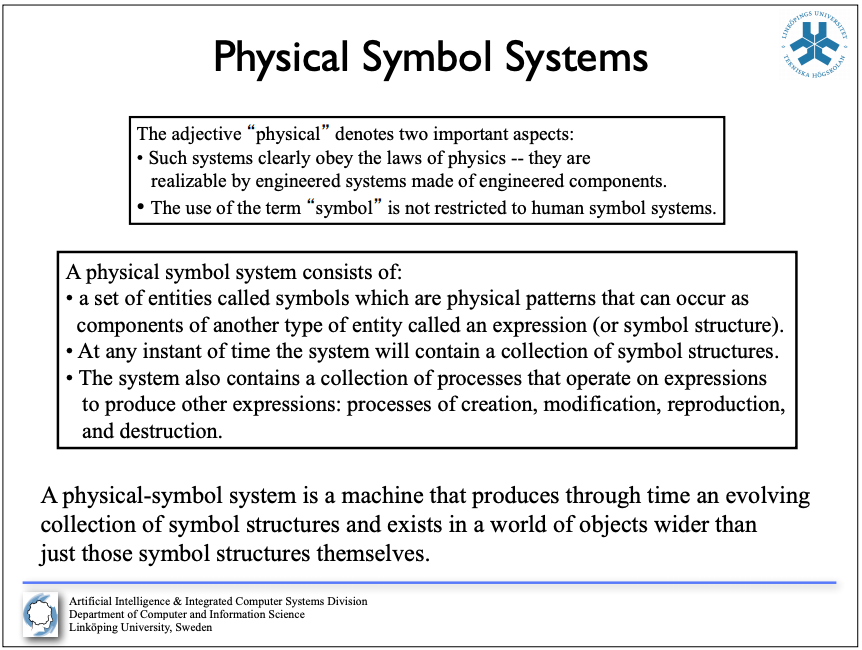
**Learning agent:**

* Has the ability to modify behavior for the better based on experience.
* It can learn new behaviors via exploration of new experiences.

**Seminar 2:**

Physical Symbol Systems

Uninformed Search



**Physical Symbol System Hypthesis:**

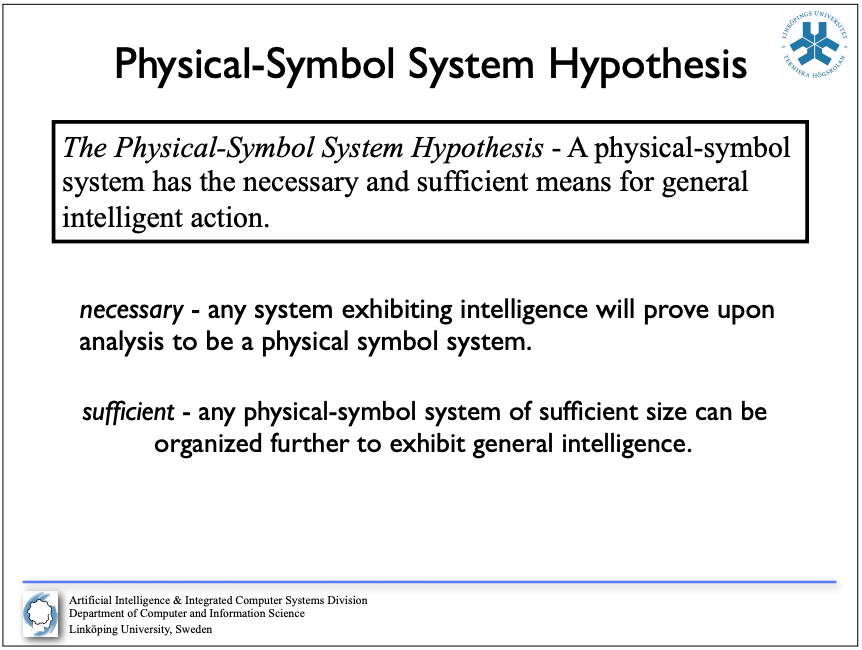
Newell and Simon are trying to lay the foundation basis for the science of AI.

What are the structural requirements for intelligence?

Can we define laws of qualitative structure for the systems being studied?

What is a symbol, that intelligence may use it, and intelligence, that it may use a symbol?

A physical-symbol system is a machine that produces through time an evolving collection of symbol structures and exists in a world of objects wider than just those symbols structures themselves.

**Designation and Interpretation:**

Two concepts central to these structures of expressions, symbols and objects:

Designation - An expression designates an object if, given the expression,

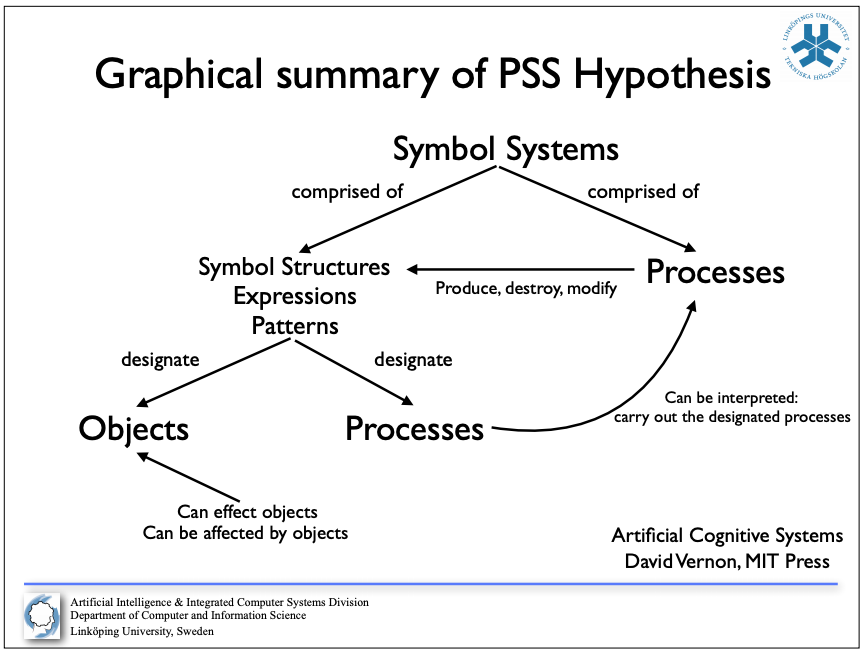
the system can either effect the object itself or behave in ways depending on

the object.

Interpretation - The system can interpret an expression if the expression

designates a process and if, given the expression, the system can carry out

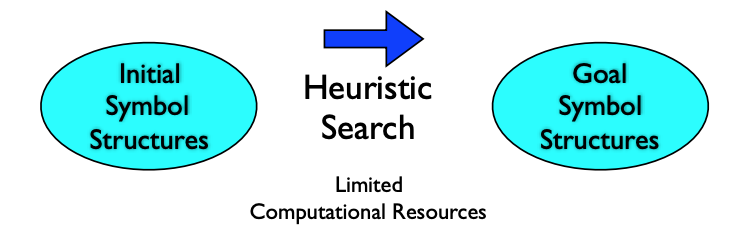
the process.

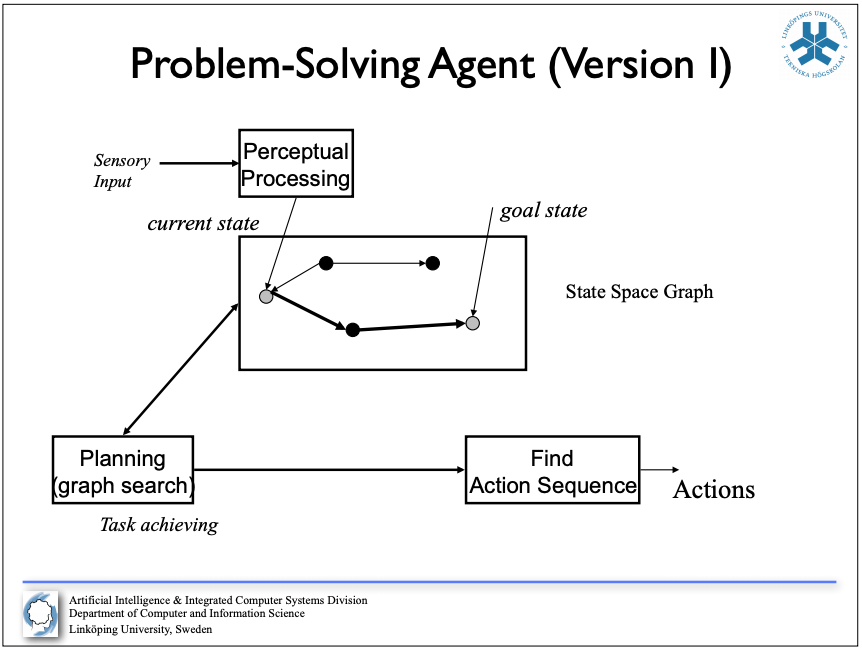


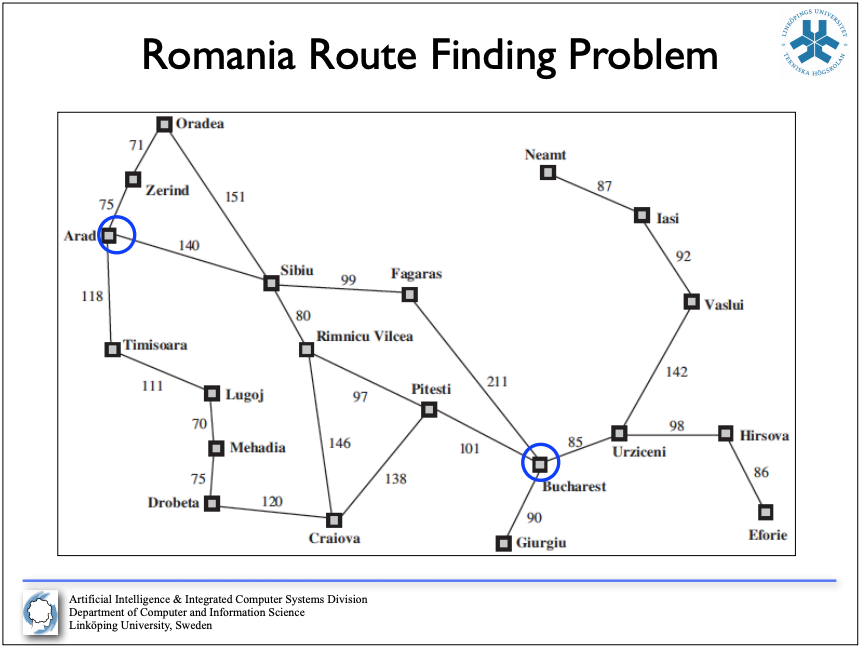
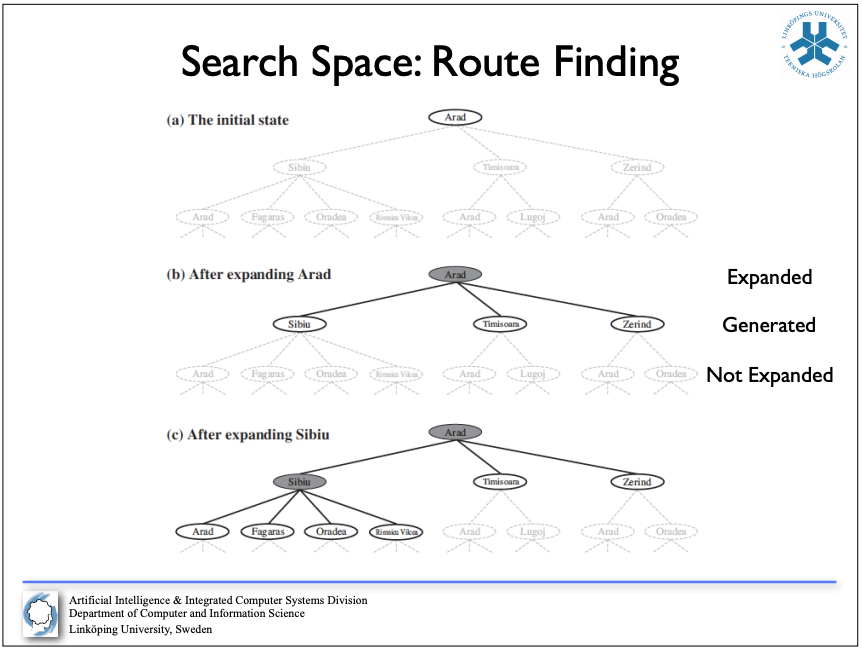
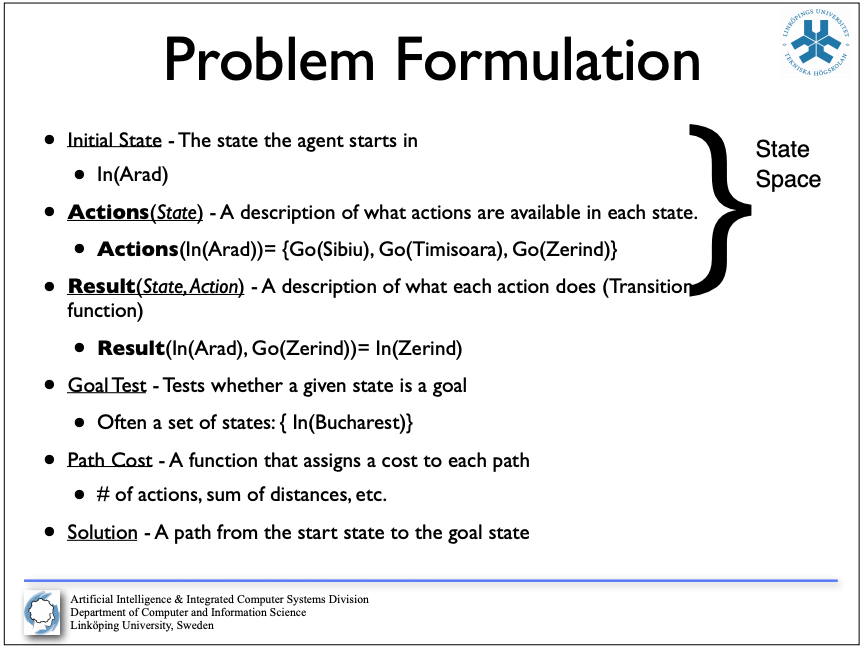
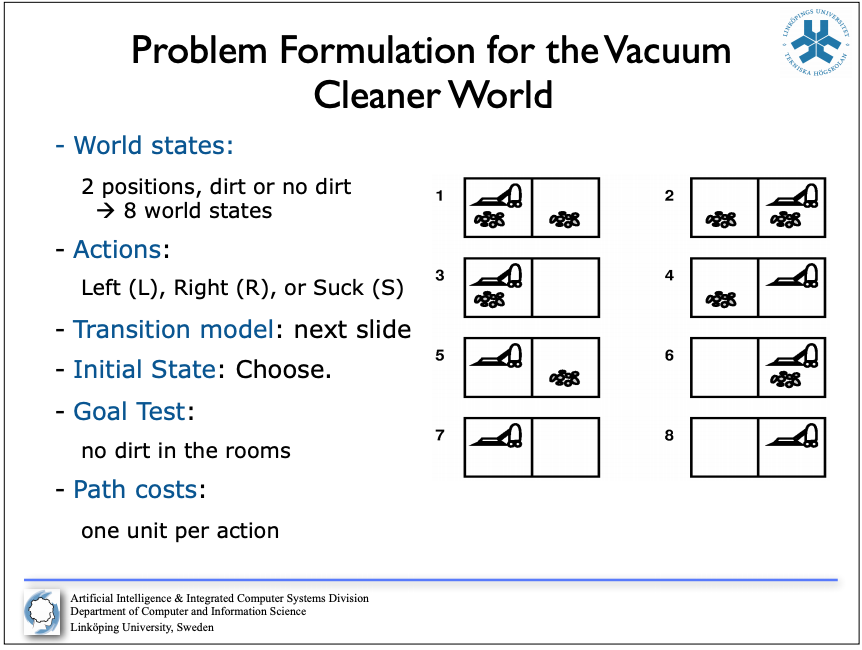
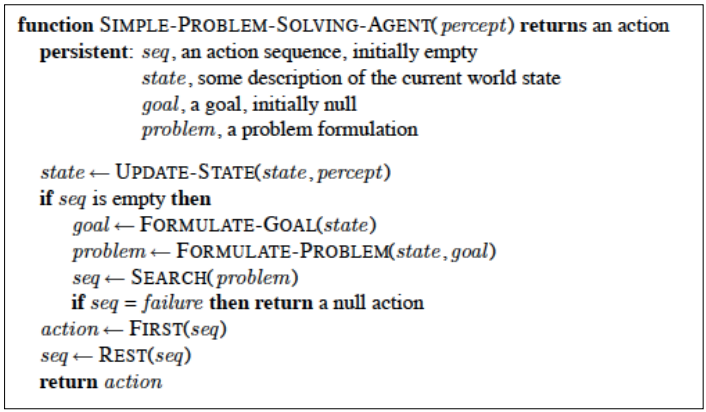
**Physical Symbol System Hypothesis:**

*Necessary* – any system exhibiting intelligence will prove upon analysis to be a physical symbol system.

*Sufficient* – any physical symbol system of sufficient size can be organized further to exhibit general intelligence.

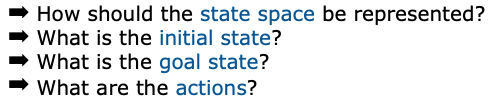
**Heuristic Search Hypothesis:**

The solutions to problems are represented as symbol structures. A physical symbol system exercises its intelligence in problem solving by search; that is, by progressively modifying structures until it produces a solution structure.

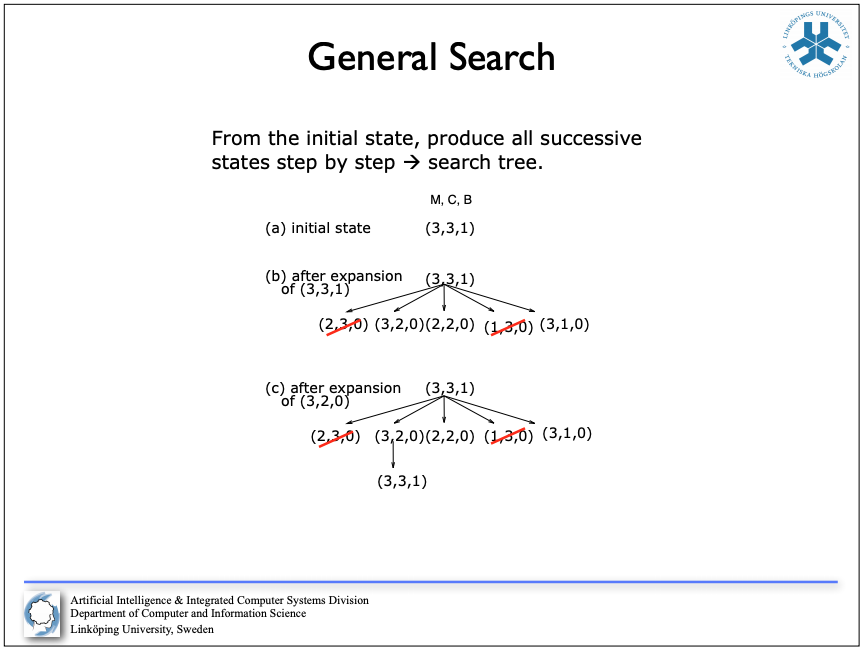
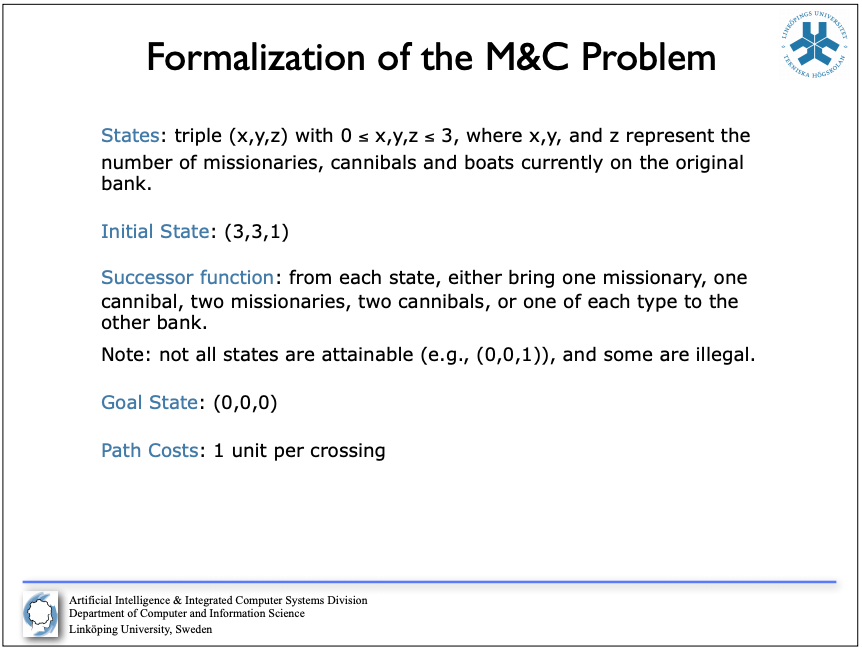
**********Simple Search:**

“Missionaries and Cannibals”

- Three missionaries and three cannibals are on one side of a river that they wish to cross.

- A boat is available that can hold at most two people.

- You must never leave a group of missionaries outnumbered by cannibals on the same bank.



Examples of real-world problems:

- Route planning, Shortest Path Problem:

- Routing video streams in computer networks, airline

travel planning, military operations planning…

- Traveling Salesperson Problem (TSP):

- A common prototype for NP-complete problems (mentioned in slides)

VLSI layout:

- Another NP-complete problem

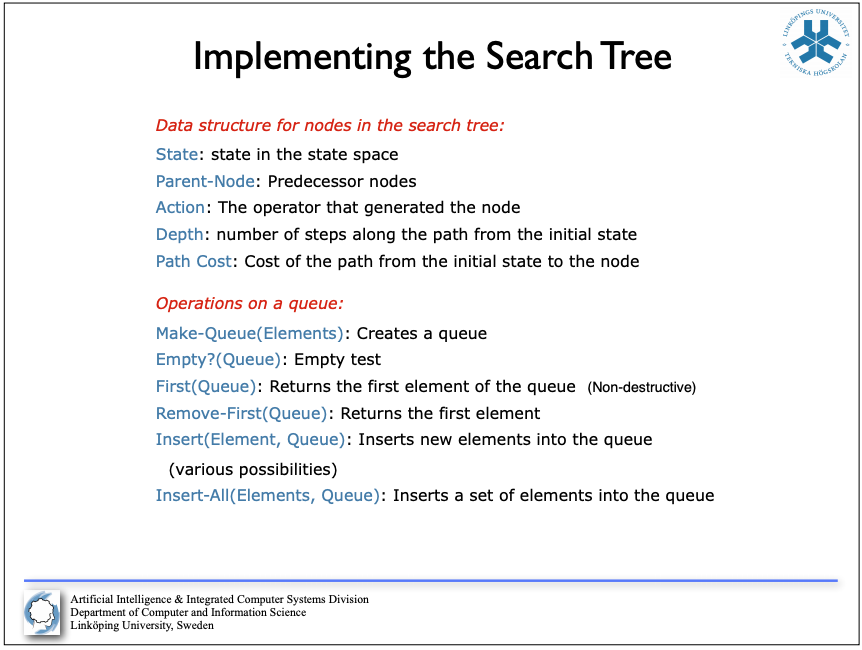
Robot Navigation (with high degrees of freedom):

- Difficulty increases quickly with the number of degrees of freedom.

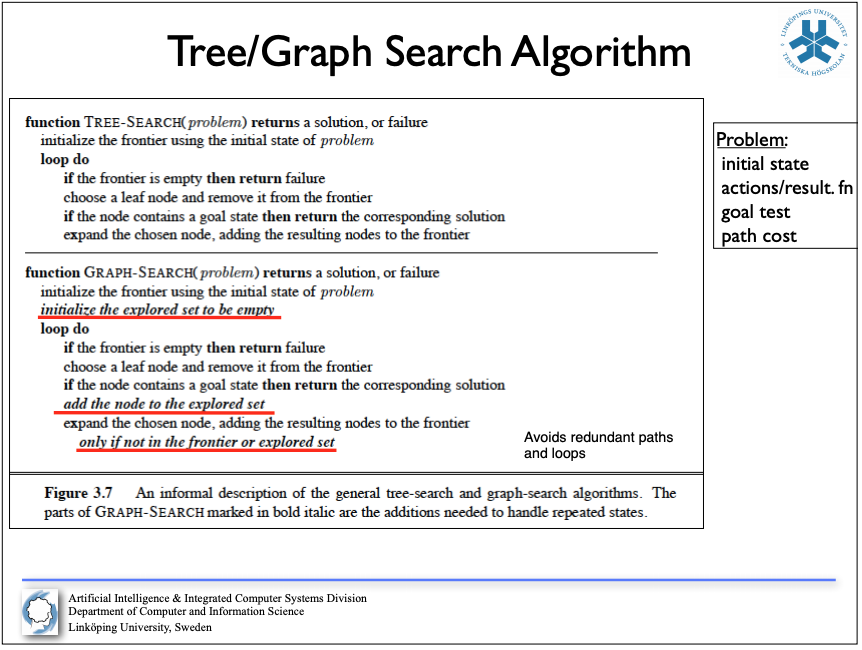
Further possible complications: errors of perception, unknown environments

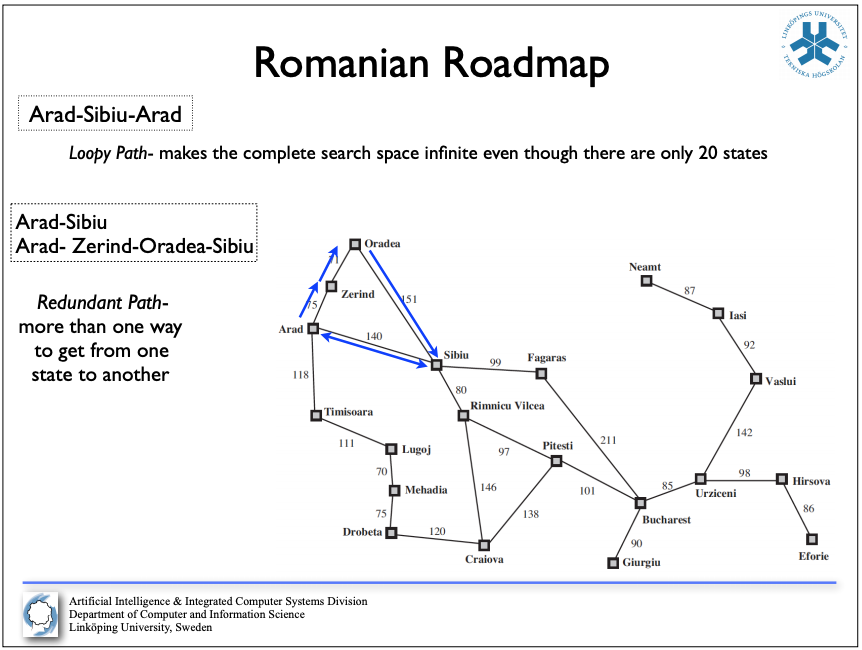
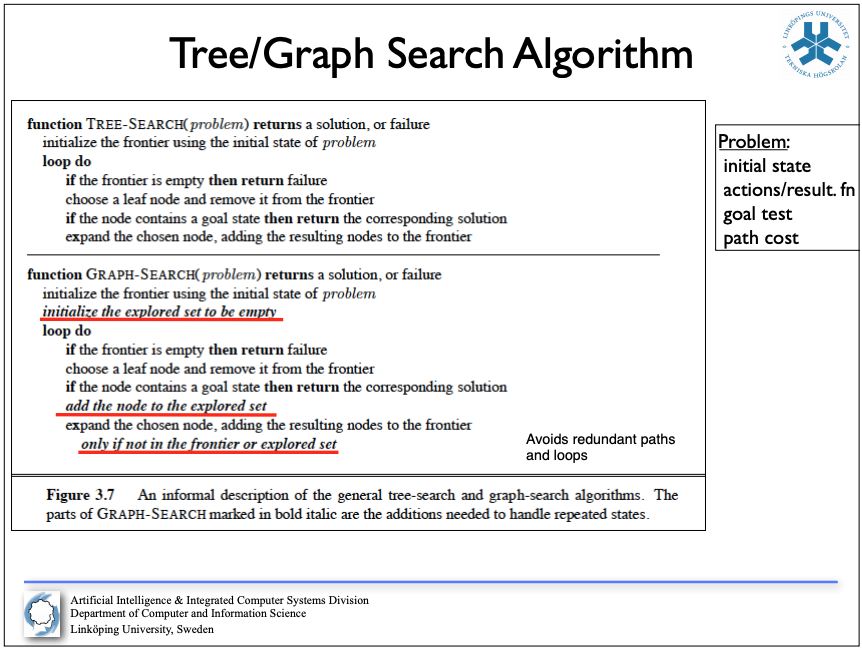
Assembly Sequencing:

- Planning of the assembly of complex objects (by robots)

States and Nodes:

A state represents a specific state of the problem (for example coordinates in a grid)



**Search Strategies:**

A strategy is defined by the order of node expansion.

Strategies are evaluated along the following dimensions:

- Completeness: does it always find a solution if one exists?

- Time Complexity: number of nodes generated/expanded

- Space Complexity: Maximum number of nodes in memory

- Optimality: does it always find a least cost solution?

Time and Space complexity are measured in terms of:

- *b* – maximum branching factor of the search tree

- *d* – depth of the least cost solution

- *m* – maximum length of any path in the state space (possibly infinite)

**Some Search Classes:**

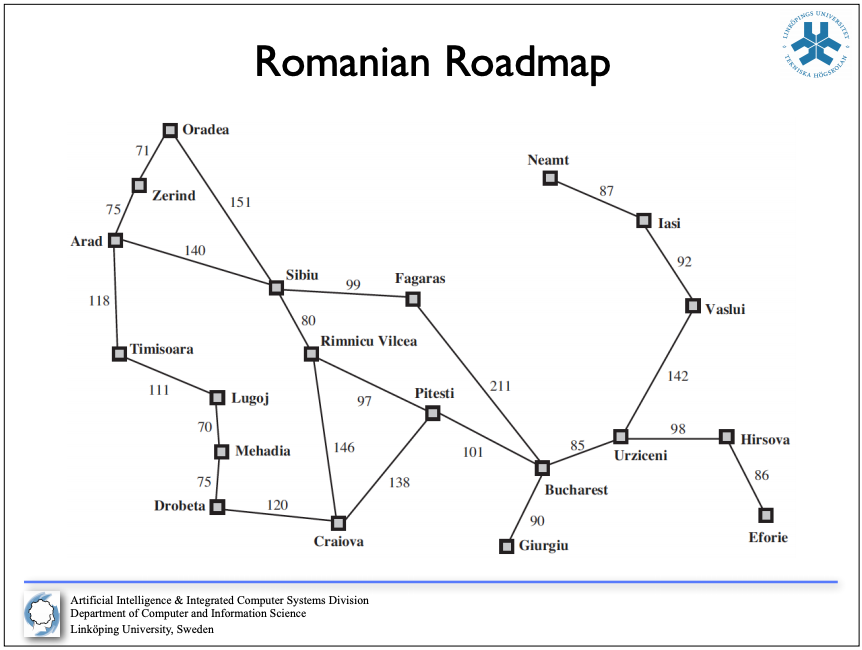
- Uninformed Search (Blind Search):

- No additional information about states besides that in the problem definition.

- Can only generate successors and compare against goal state.

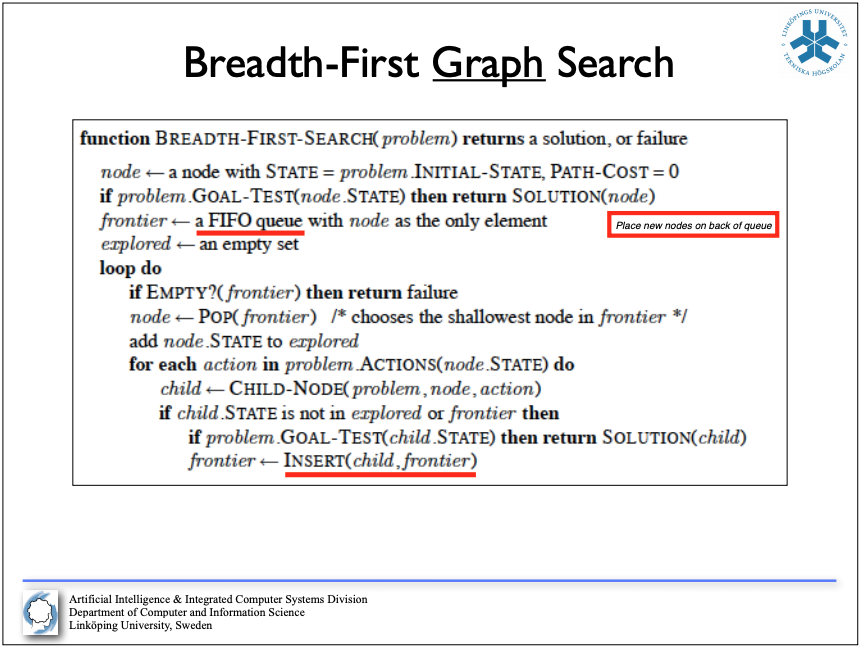
- Some examples:

- Breadth First Search, Depth First Search, Iterative Deepening DFS

- Informed Search (Heuristic Search):

- Strategies have additional information whether non-goal states are more promising than others

- Some examples:

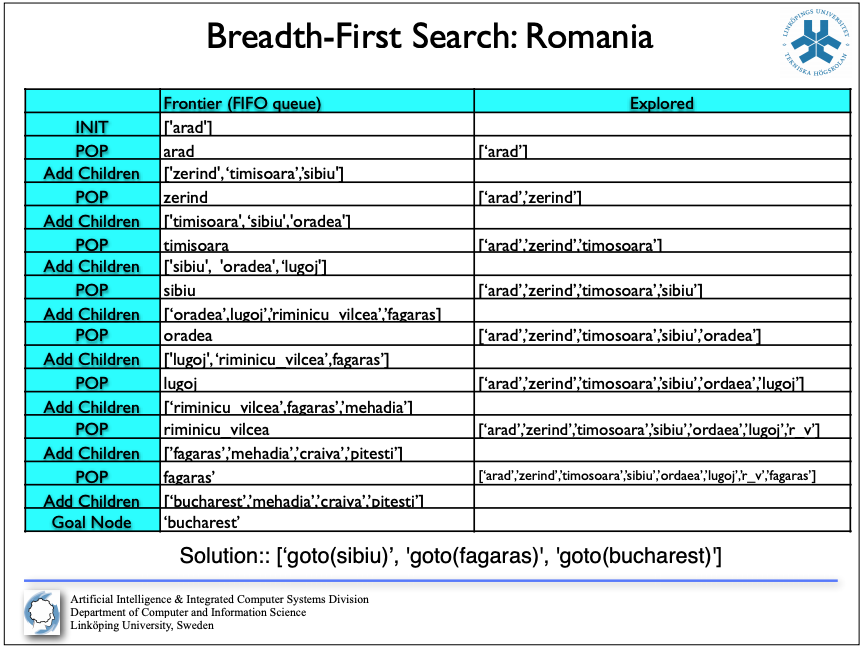
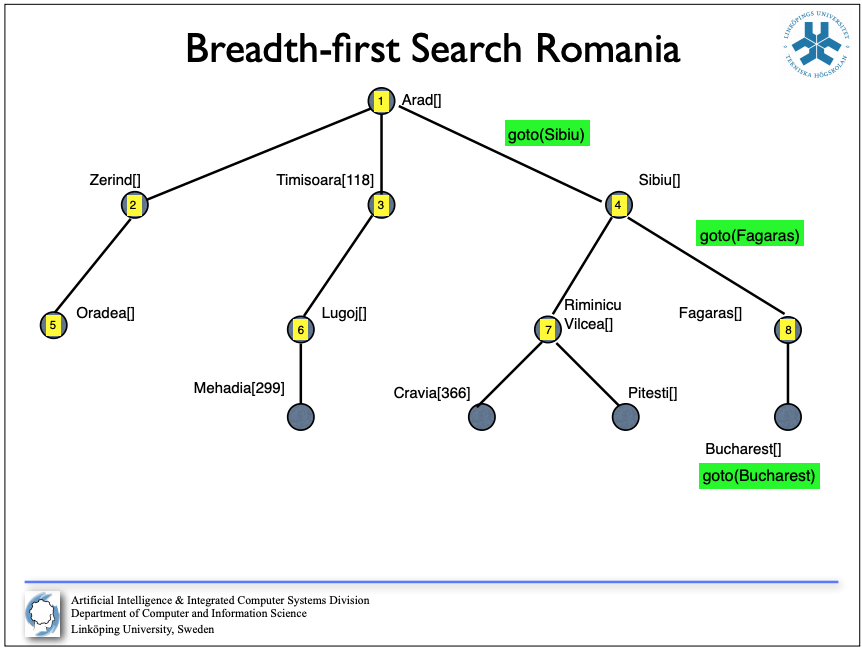
 - Greedy Best First Search, A\* Search

**BFS:**

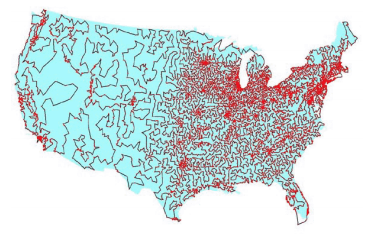
Search consists of a frontier (FIFO queue for BFS) and an explored vector.

Expand each “depth” of the tree until the goal-node is found, i.e. add nodes to the frontier.

When searching further in the tree pop the node at the front of the queue and add that node to the explored vector.

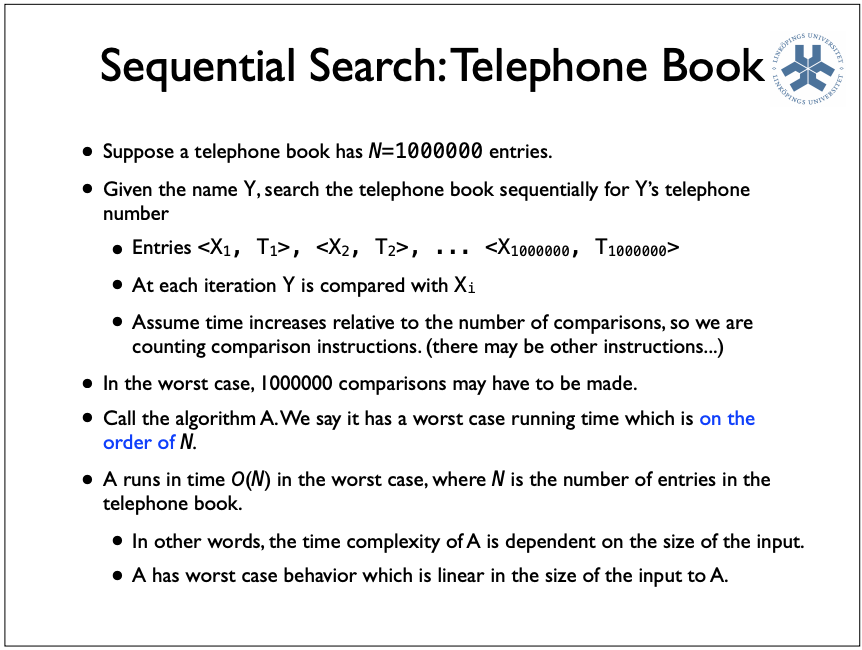
When found, trace back to the starting node and return each node taken from goal- to starting-node. (Picture in previous page for pseudo-code).

**Computational Complexity Theory:**

Traveling Salesman Problem:

- The traveling salesman problem is one of the most intensively studied problems in computational mathematics.

- A traveling salesman has *n* number of cities to visit. He wants to know the shortest route which will allow him to visit all cities one time and return to his starting point.

- Solving this problem becomes much harder as the number of cities increase; figure to the right shows solution to 13509 cities in the US having more than 500 residents.

**Big-*O* Notation**