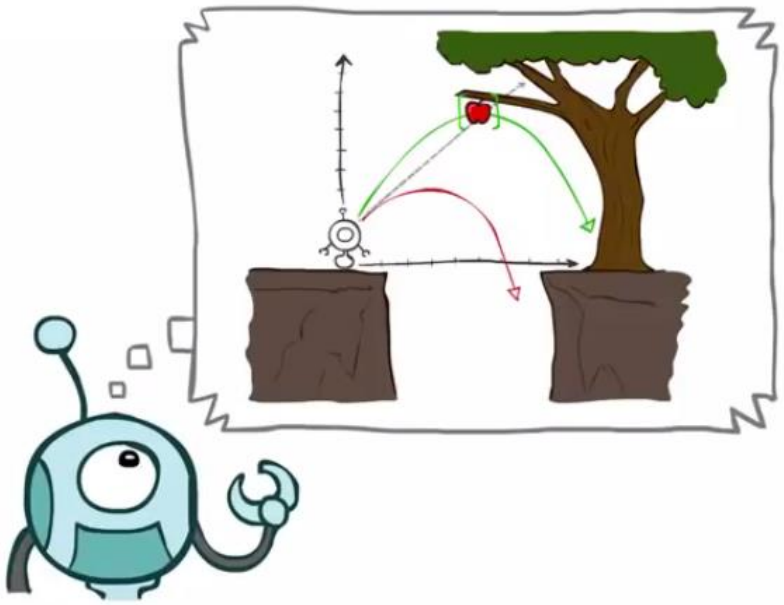
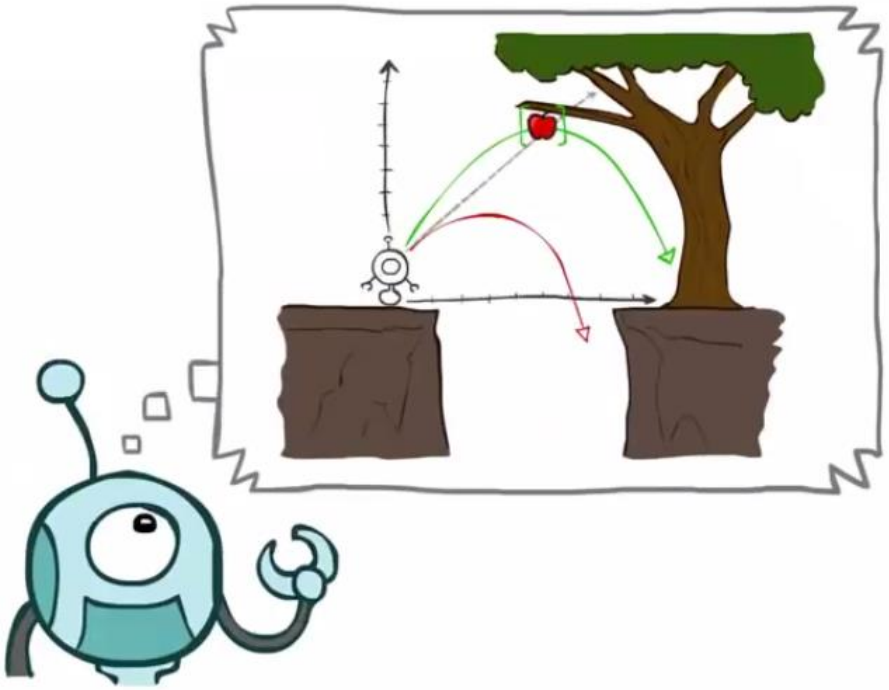
Rational Agents (Chapter 2)

Today

- Agents that Plan Ahead
- Search Problems
- Uninformed Search Methods
  - Depth-First Search
  - Breadth-First Search
  - Uniform-Cost Search

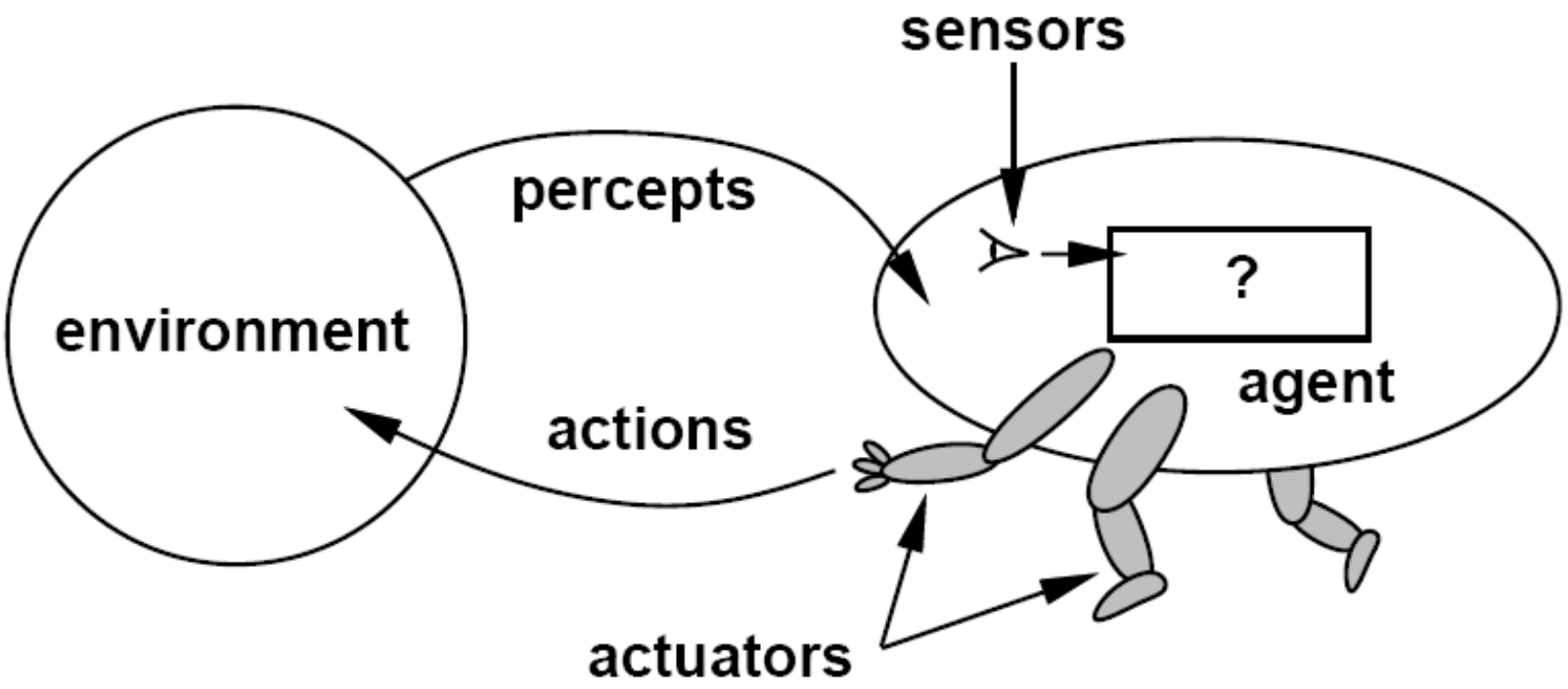


Agents that Plan



Agents

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



Agents

What will be the component of Human Agent?

Human agent:

Sensors: eyes, ears, and other organs for sensors;

Actuators: hands, legs, mouth, and other body parts.

What will be the component of Robotic Agent?

Robotic agent:

Sensors: cameras, touch sensor and infrared range finders

Actuator: various motors for actuators

Agents and environments

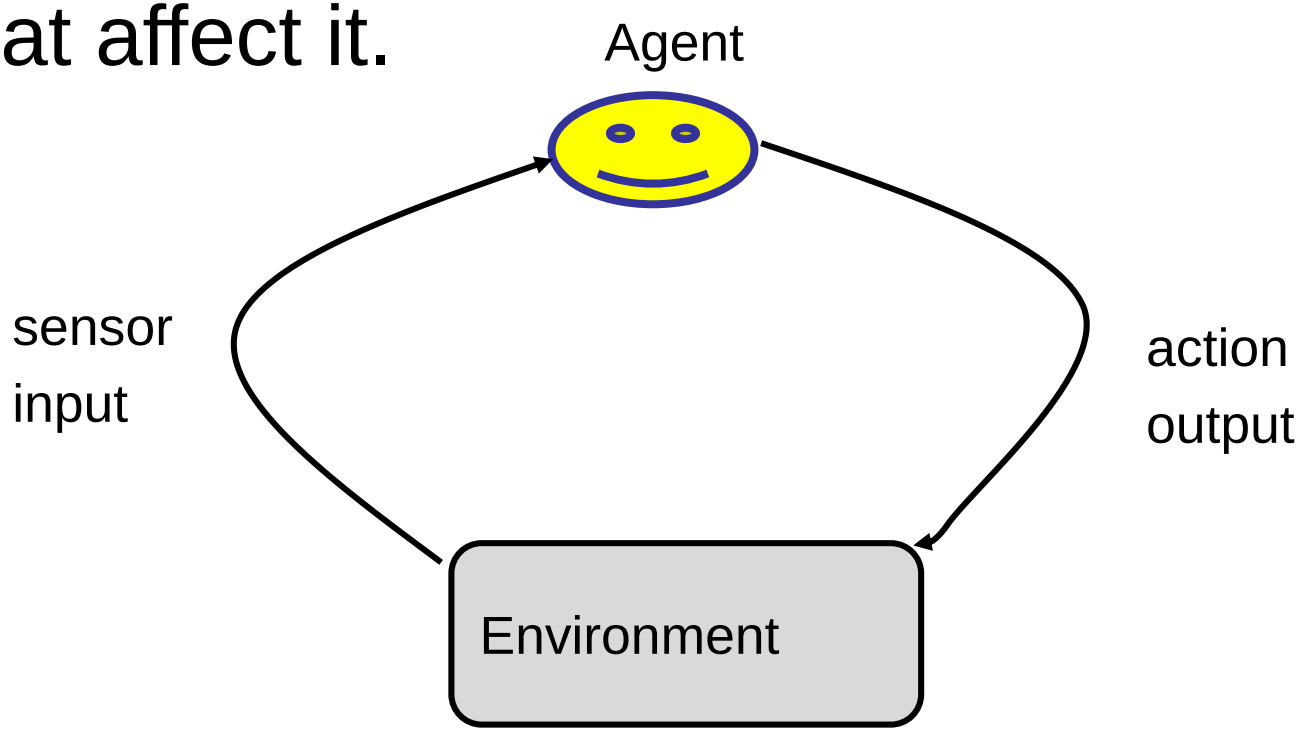
- The agent function maps from percept histories to actions:

$$[f: \mathcal{P}^{\star} \rightarrow \mathcal{A}]$$

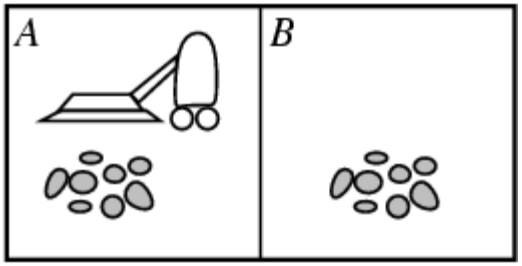
- The agent program runs on the physical architecture to produce *f.*
- agent comprised of : architecture + program

Agents & Environments

- The agent takes sensory input from its environment, and produces as output actions that affect it.



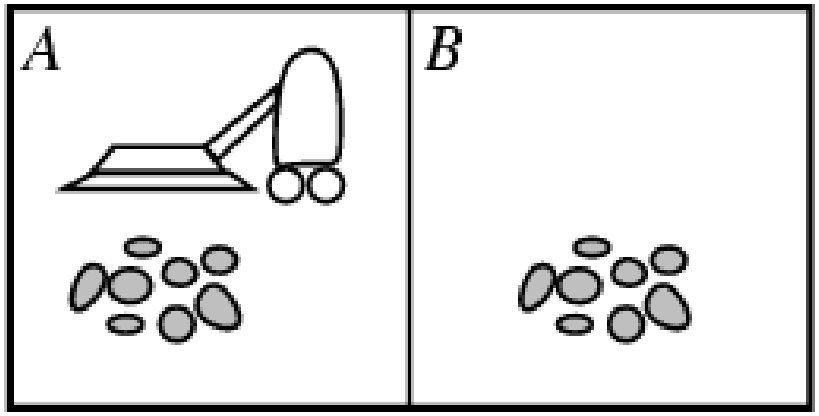
Vacuum-cleaner world



- Percepts: location and contents, e.g., [A,Dirty]
- Actions: *Left*, *Right*, Clean, *NoOp*

Example: Vacuum-Agent

- **Percepts:**
  Location and status,
  e.g., [A,Dirty]
- **Actions:**
  Left, Right, Suck, NoOp



function Vacuum-Agent([location,status]) returns an action

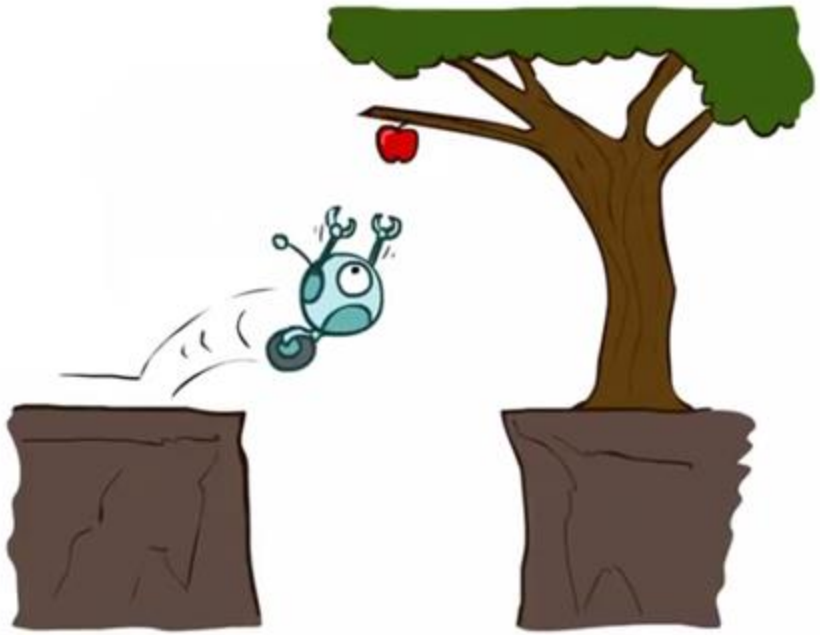
- *if* status = Dirty *then* return Suck
- *else if* location = A *then* return Right
- *else if* location = B *then* return Left

A vacuum-cleaner agent

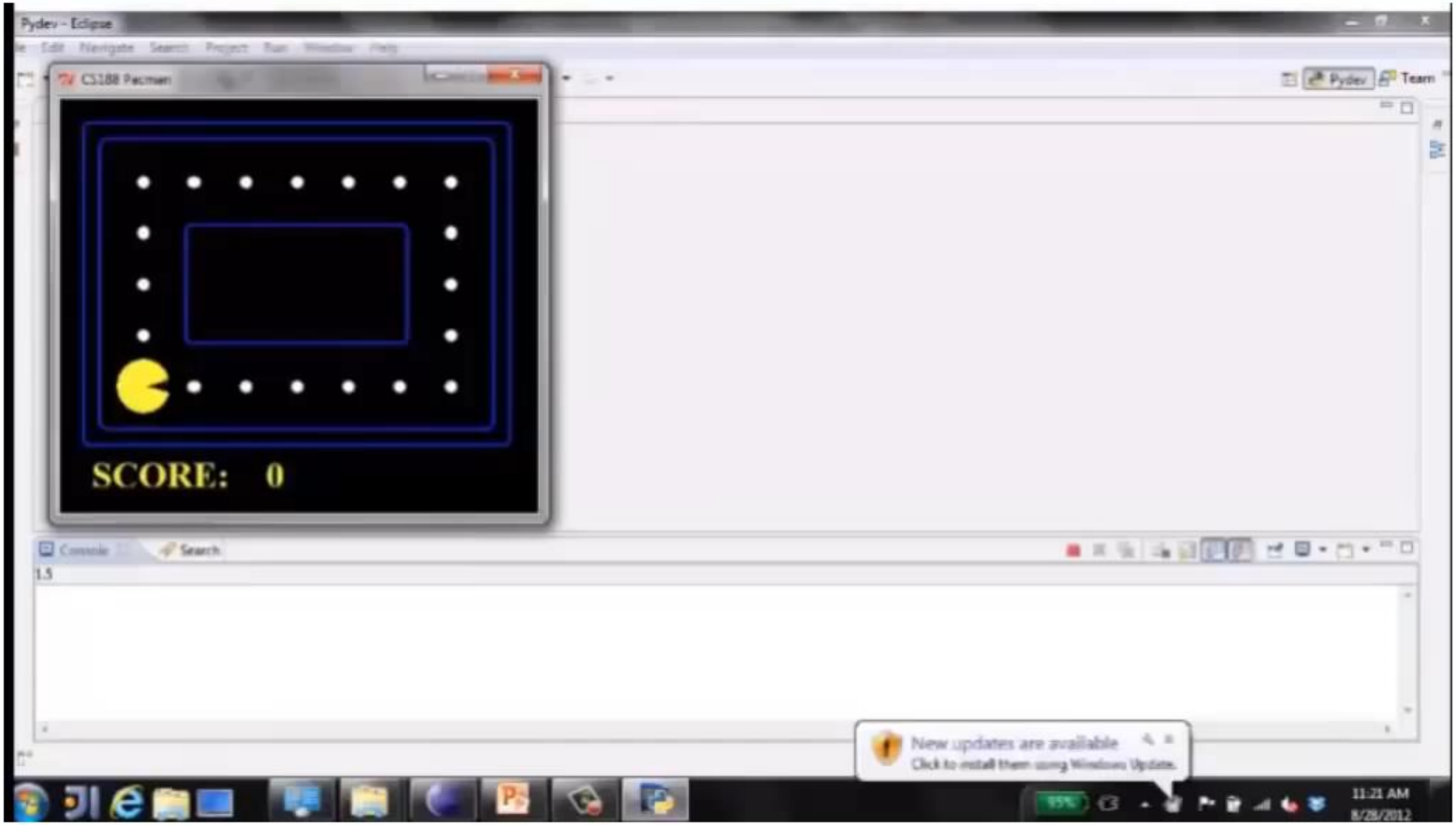
Percept Sequence	Action
[A, Fresh]	Right
[A,Dirty]	Clean
[B, Fresh]	Left
[B, Dirty]	Clean
[A, Fresh], [A, Fresh]	Right
[A, Fresh], [A,Dirty]	Clean
….	…..
[A, Fresh], [A, Fresh], [A, Fresh]	Right
[A, Fresh], [A, Fresh], [A, Dirty]	Clean
…..	….

Reflex Agents

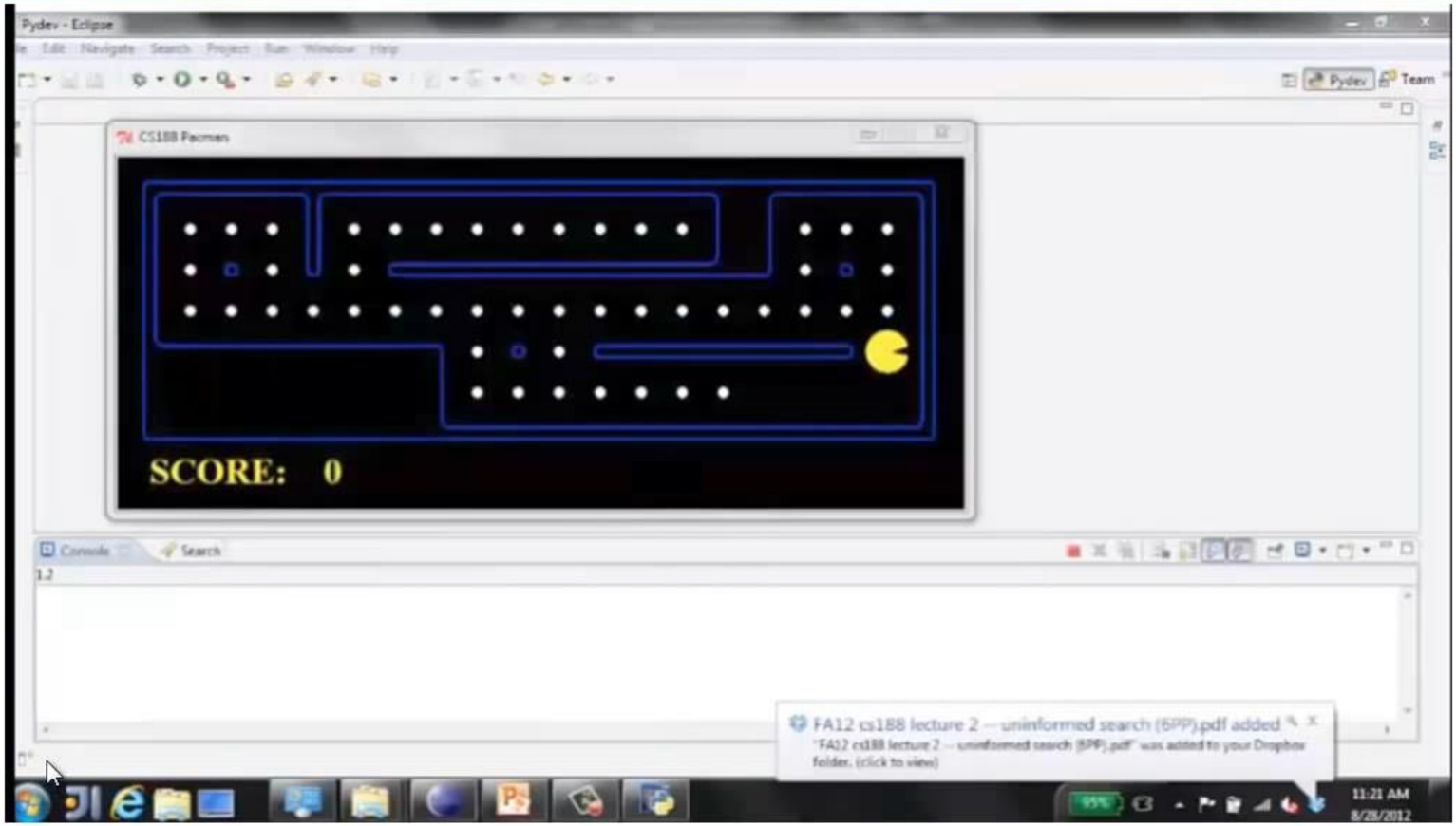
- Reflex agents:
  - Choose action based on current percept (and maybe memory)
  - May have memory or a model of the world's current state
  - Do not consider the future consequences of their actions
  - Consider how the world IS
- Can a reflex agent be rational?



Demo: Reflex / Optimal Loop

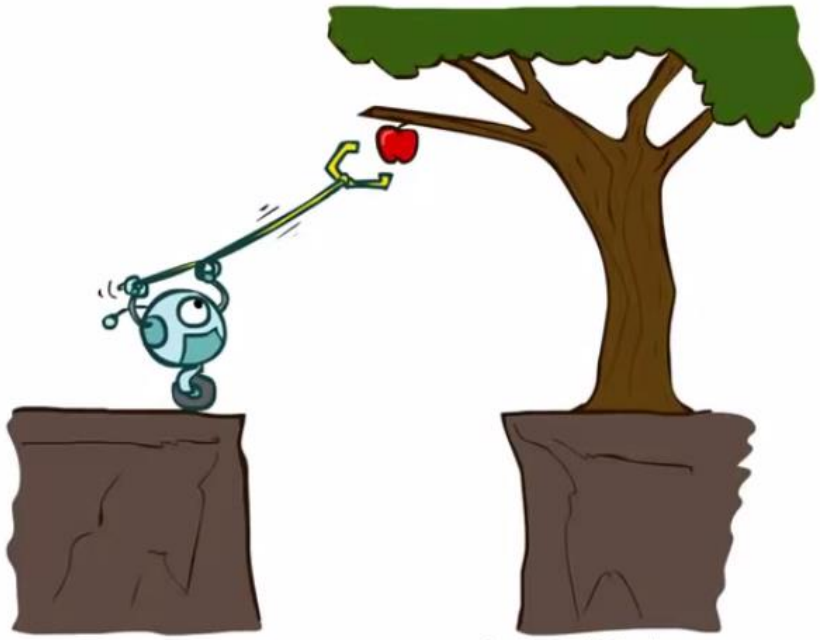


Demo: Reflex /

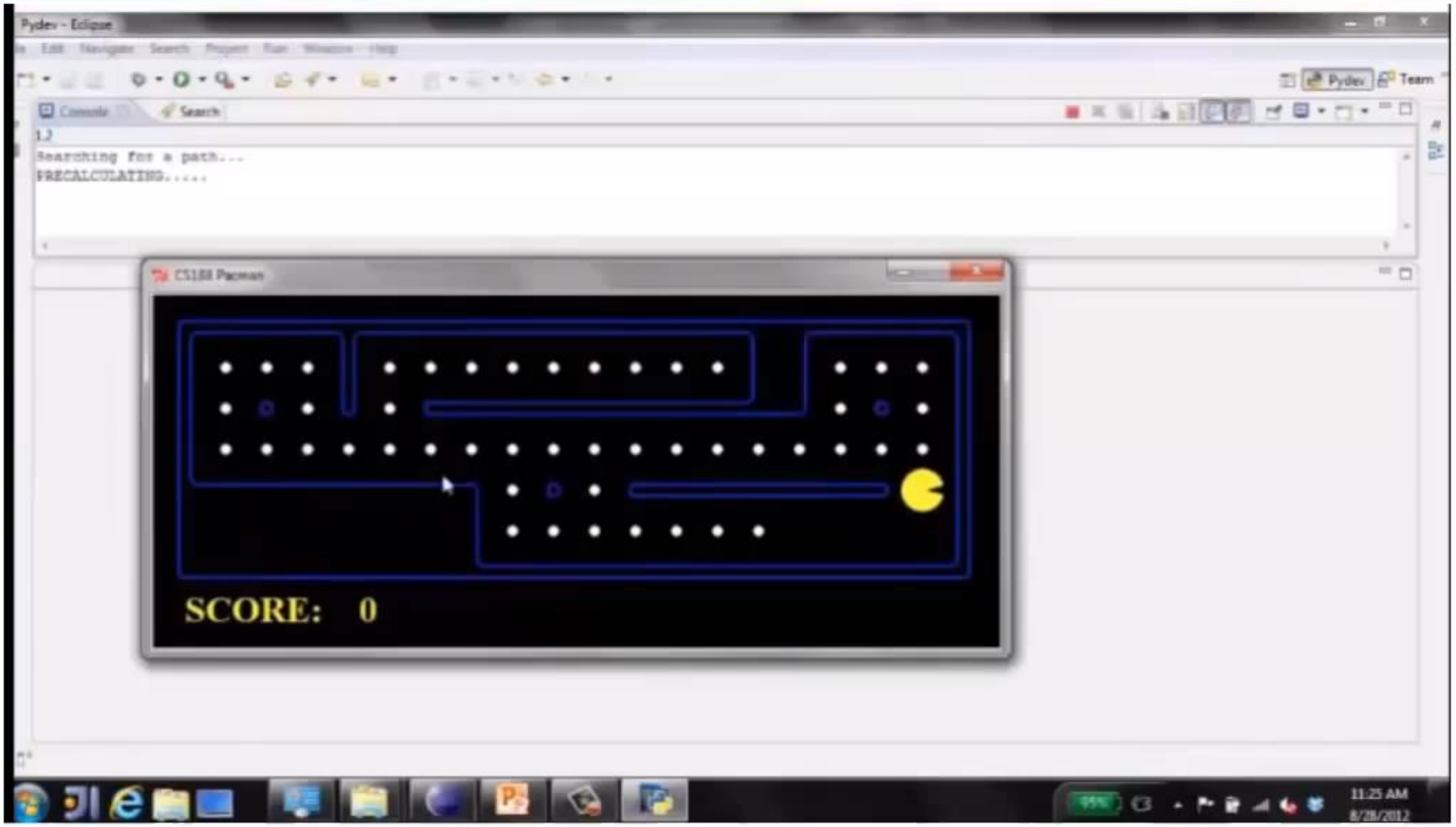

Pydev - Eclipse
CS188 Pacmen
SCORE: 0
Console
Search
FA12 cs188 lecture 2 -- uninformed search (6PP).pdf added

Planning Agents

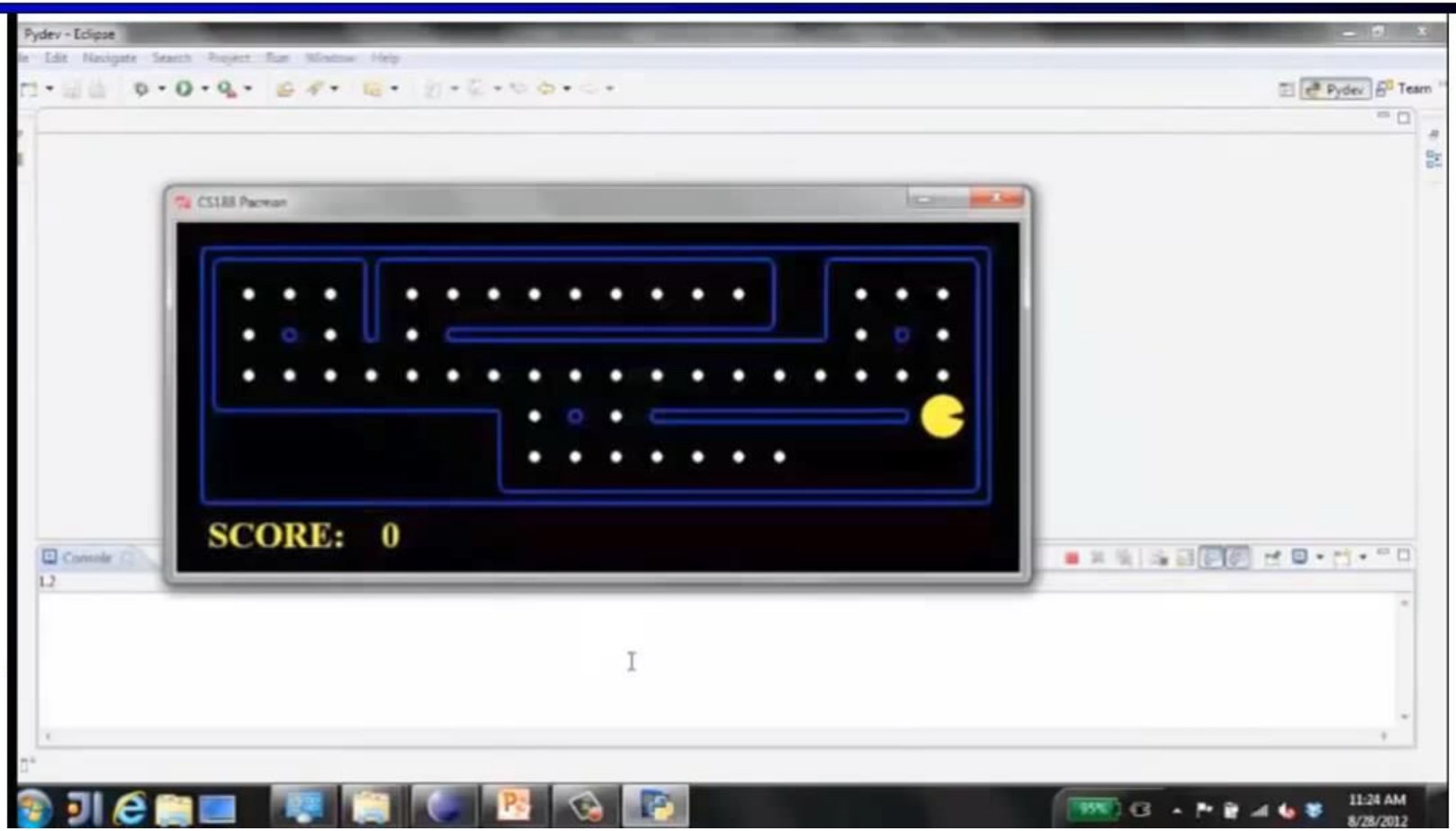
- Planning agents:
  - Ask "what if"
  - Decisions based on (hypothesized) consequences of actions
  - Must have a model of how the world evolves in response to actions
  - Must formulate a goal (test)
  - Consider how the world WOULD BE
- Optimal vs. complete planning
- Planning vs. replanning



Demo: Plan Slow (“mastermind”)



Demo: Plan Fast ("replanning")



Rational agents (1/3)

- An agent should strive to "do the right thing", based on what it can perceive and the actions it can perform. The right action is the one that will cause the agent to be most successful.

- Performance measure: An objective criterion for success of an agent's behavior.

- E.g., performance measure of a vacuum-cleaner agent could be amount of dirt cleaned up, amount of time taken, amount of electricity consumed, amount of noise generated, etc.

Rational agents (2/3)

- Rational Agent: For each possible percept sequence,
- a rational agent should select an action that is expected to maximize its performance measure,
- given the evidence provided by the percept sequence and whatever built-in knowledge the agent has.

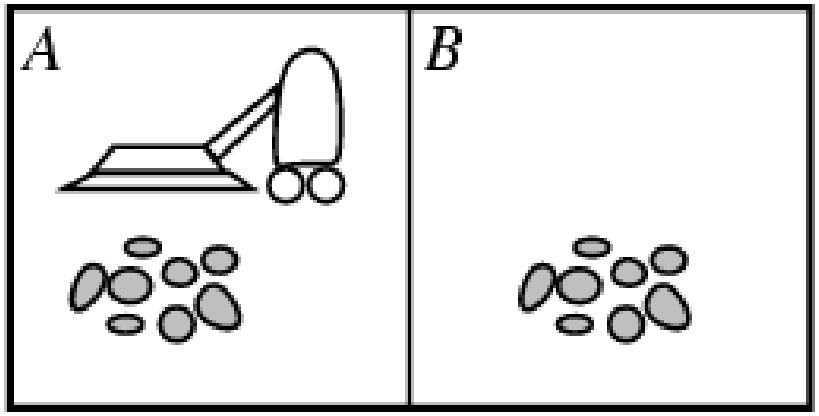
Rational agents (3/3)

- An agent is autonomous if its behavior is determined by its own experience

  (with ability to learn and adapt).

Back to Vacuum-Agent

- **Percepts:**
  Location and status,
  e.g., [A,Dirty]
- **Actions:**
  Left, Right, Suck, NoOp



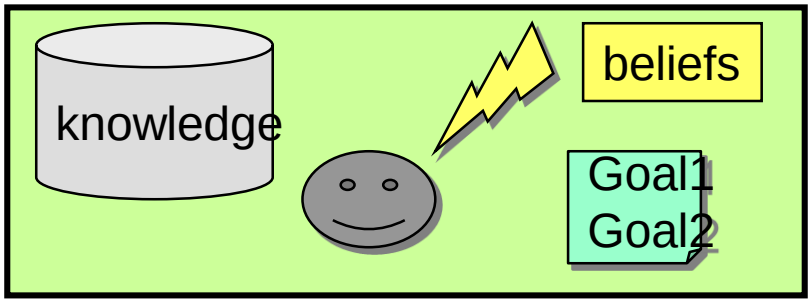
function Vacuum-Agent([location,status]) returns an action

- *if* status = Dirty *then* return Suck
- *else if* location = A *then* return Right
- *else if* location = B *then* return Left

- Is this agent rational?
  - Depends on performance measure, environment properties

Agent Characterisation

- An agent is responsible for satisfying specific *goals*. There can be different types of goals such as achieving a specific status (defined either exactly or approximately), keeping certain status, optimizing a given function (e.g., utility), etc.

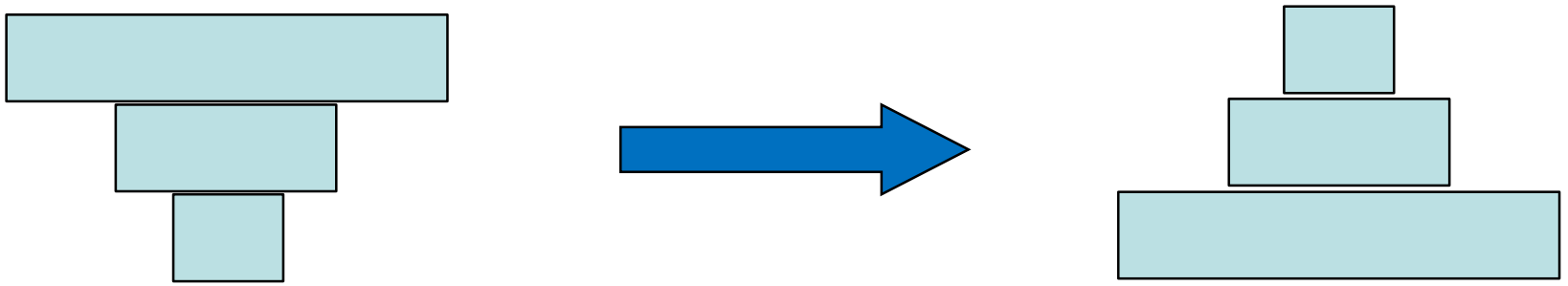


- The *state* of an agent includes state of its *internal environment* + state of *knowledge* and *beliefs* about its *external environment*.

Goal I (achieving exactly defined status)

Initial State

Goal



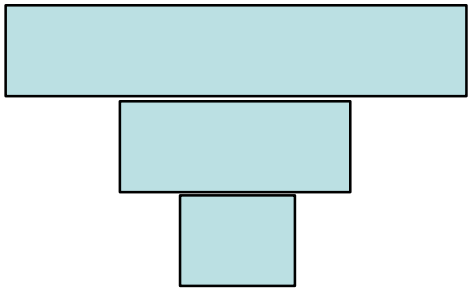
Goal II (achieving constrained status)

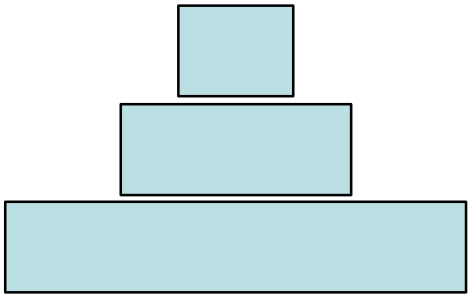
Goal

Constraint:

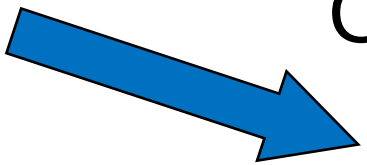
"*The smallest in on top*"

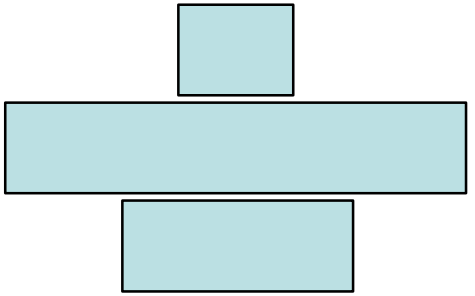
Initial State





OR

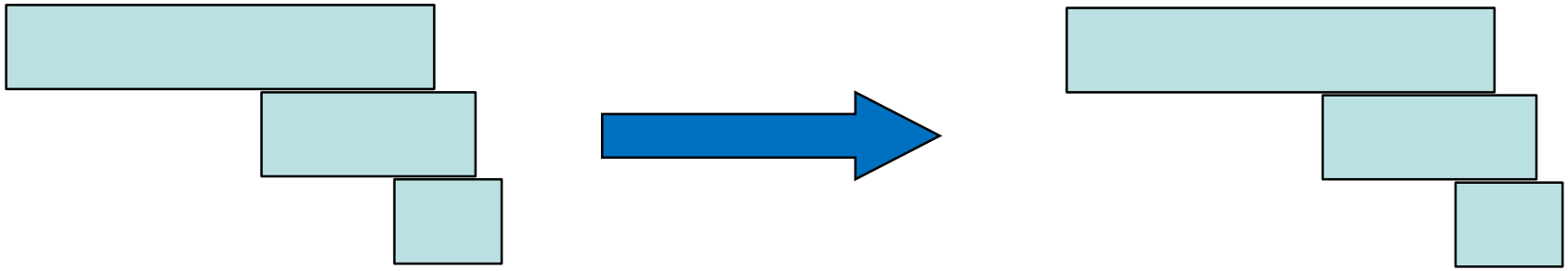




Goal III (continuously keeping instable status)

Initial State

Goal



Goal IV (maximizing utility)

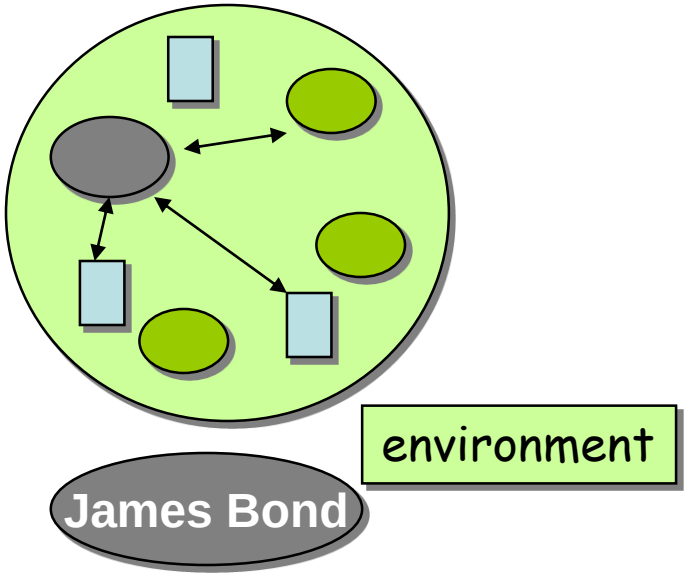
Initial State

Goal:

The basket filled with mushrooms that can be sold for maximum possible price

Situatedness

- An agent is *situated* in an *environment*, that consists of the objects and other agents it is possible to interact with.



- An agent has an *identity* that distinguishes it from the other agents of its environment.

Specifying the task environment

- **PEAS: Performance measure, Environment, Actuators, Sensors**
- **P:** a function the agent is maximizing (or minimizing)
  - Assumed given
- **E:** a formal representation for *world states*
  - For concreteness, a tuple (var_1=val_1, var_2=val_2, … ,var_n=val_n)
- **A:** actions that change the state according to a *transition model*
  - Given a state and action, what is the successor state (or distribution over successor states)?
- **S:** observations that allow the agent to infer the world state
  - Often come in very different form than the state itself
  - E.g., in tracking, observations may be pixels and state variables 3D coordinates

PEAS Example: Autonomous taxi

- Performance measure
  - Safe, fast, legal, comfortable trip, maximize profits
- Environment
  - Roads, other traffic, pedestrians, customers
- Actuators
  - Steering wheel, accelerator, brake, signal, horn
- Sensors
  - Cameras, LIDAR, speedometer, GPS, odometer, engine sensors, keyboard

Another PEAS example: Spam filter

- Performance measure
  - Minimizing false positives, false negatives
- Environment
  - A user's email account, email server
- Actuators
  - Mark as spam, delete, etc.
- Sensors
  - Incoming messages, other information about user's account

Another PEAS example

Agent: Medical diagnosis system

- Performance measure: Healthy patient, minimize costs, fewer lawsuits.
- Environment: Patient, hospital, staff.
- Actuators: Screen display (questions, tests, diagnoses, treatments, referrals).
- Sensors: Keyboard (entry of symptoms, findings, patient's answers).

Another PEAS example

Agent: Part-picking robot

- Performance measure: Percentage of parts in correct bins
- Environment: Conveyor belt with parts, bins
- Actuators: Jointed arm and hand
- Sensors: Camera, joint angle sensors

Another PEAS example

Agent: Interactive English tutor

- Performance measure: Maximize student's score on test
- Environment: Set of students
- Actuators: Screen display (exercises, suggestions, corrections)
- Sensors: Keyboard

Thank You