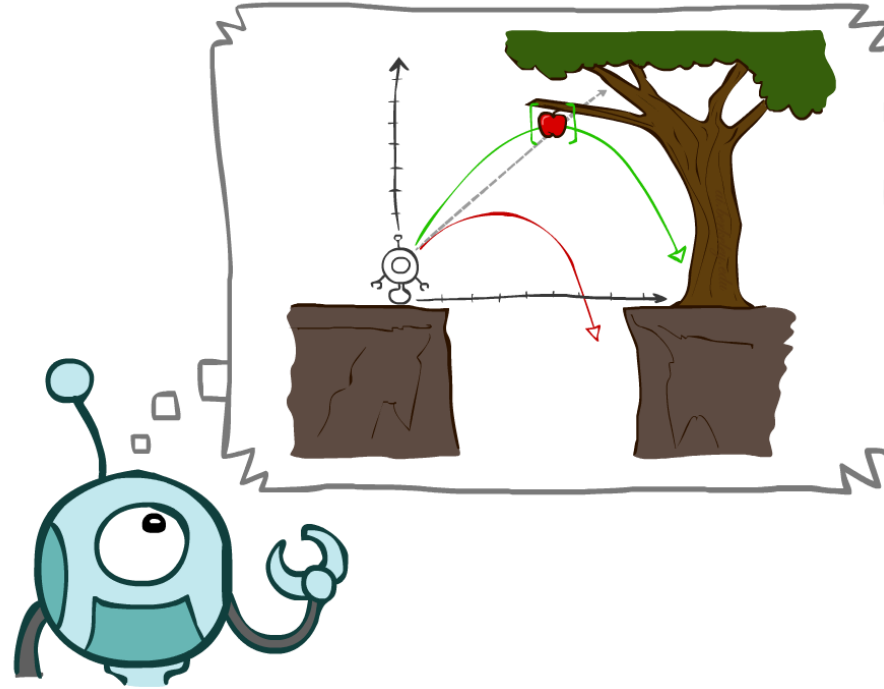


Yapay Zeka

Akıllı Ajanlar

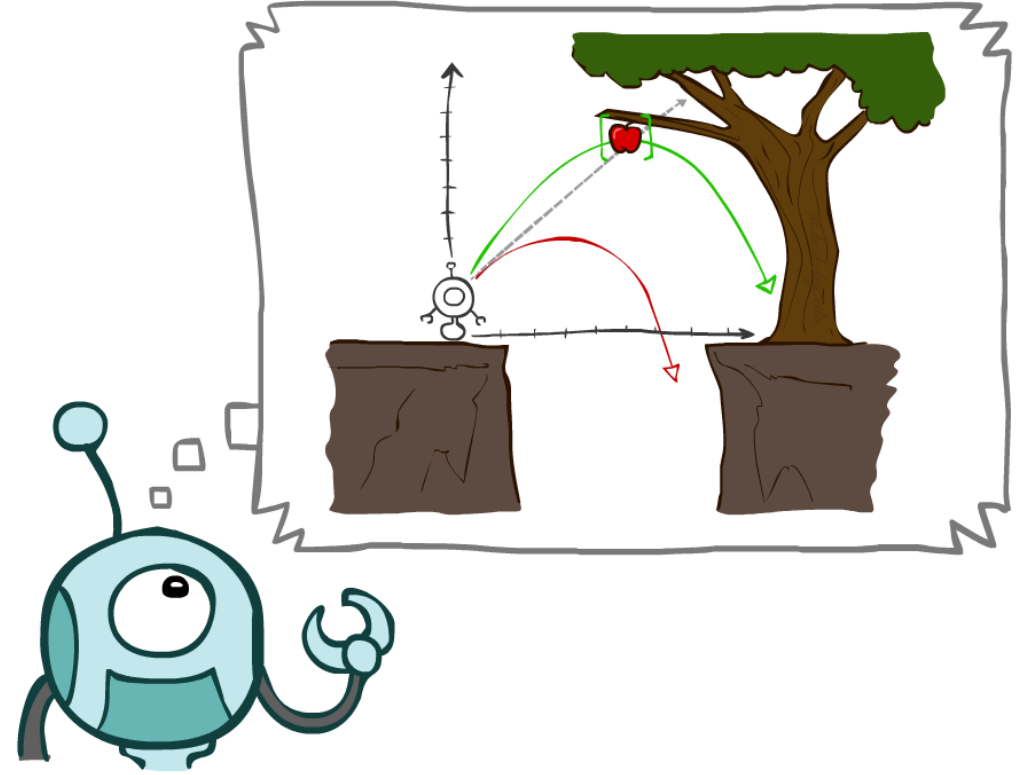


Sevinç İlhan Omurca / Fidan Kaya Gülağz

Kocaeli Üniversitesi

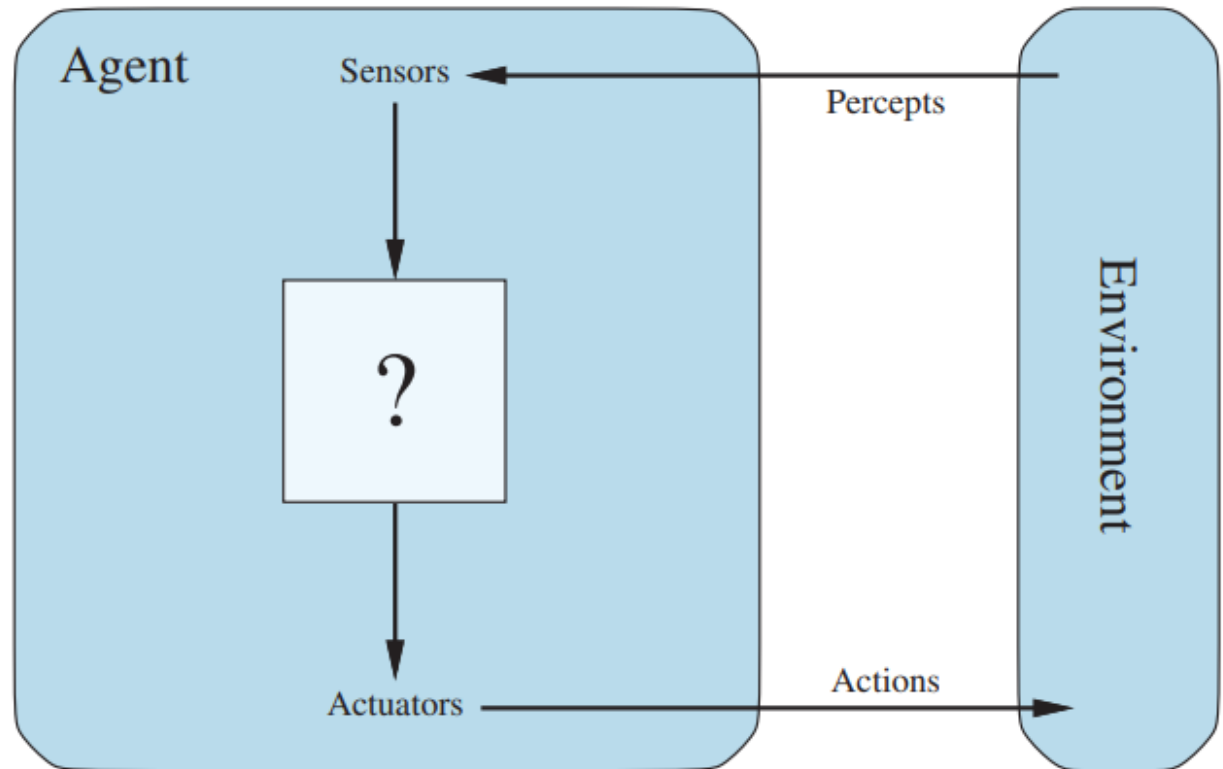
Bu Hafta

- Ajanlar ve Çevre
- Rasyonel Ajan
- Görev Ortamı (Environment Task)
- Ajan Türleri



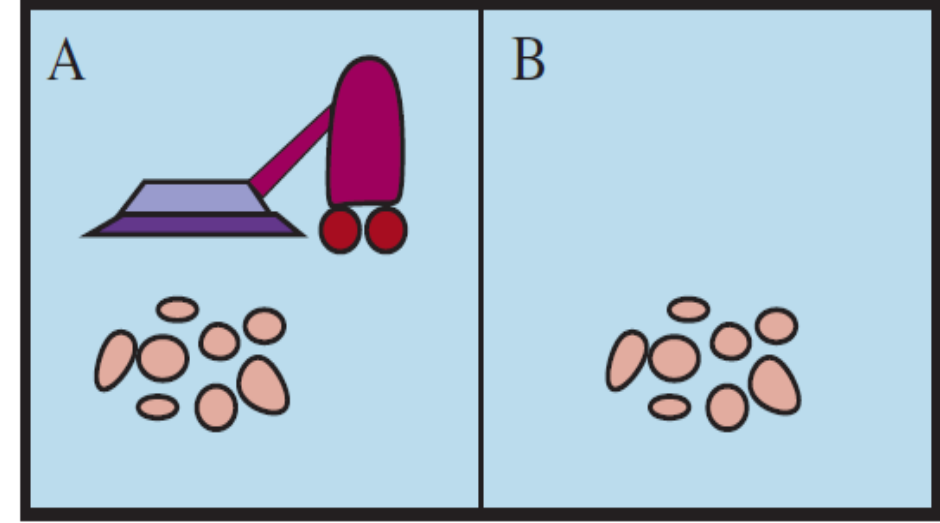
Ajanlar ve Çevre

- Ajan (Agent)
- Sensörler (Sensors)
- Harekete Geçiriciler (Actuators)
- Ajan Fonksiyonu
 - $f : P^* \rightarrow A$



Rasyonel Ajan (İyi Davranış)

- Rasyonel bir ajan doğru olanı yapar!
- Bir ajanı iyi ya da kötü yapan nedir?
- Şekildeki ajan mantıklı bir ajan mıdır?
 - Performans ölçüsü?
 - Çervesel Özellikler (Sensors)?
 - Harekete Geçiriciler (Actuators)?



Percept sequence	Action
[A, Clean]	Right
[A, Dirty]	Suck
[B, Clean]	Left
[B, Dirty]	Suck
[A, Clean], [A, Clean]	Right
[A, Clean], [A, Dirty]	Suck
⋮	⋮
[A, Clean], [A, Clean], [A, Clean]	Right
[A, Clean], [A, Clean], [A, Dirty]	Suck
⋮	⋮

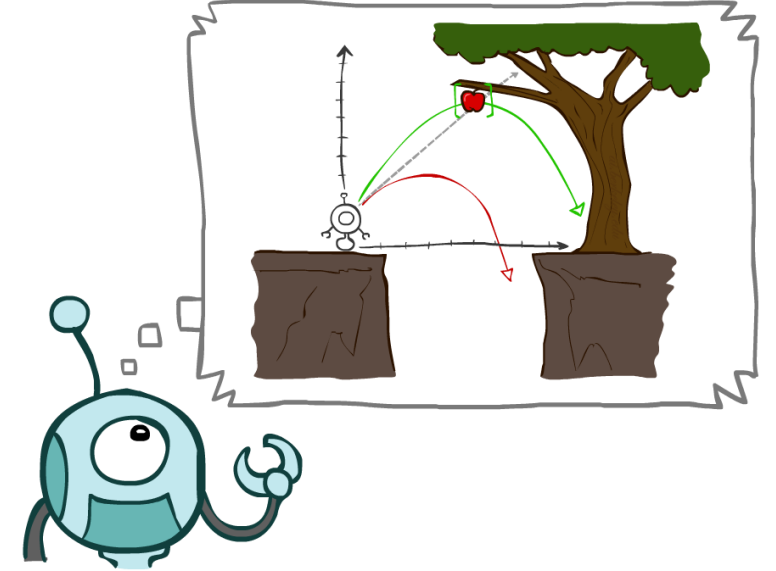
Rasyonel Ajan (İyi Davranış)

- Rasyonel hareket 4 duruma bağlıdır.

- Performans ölçęęi
- Çevre ile ilgili ön bilgi
- Ajanın gerçekleştirebileceęi eylemler
- Ajanın algı geçmişı

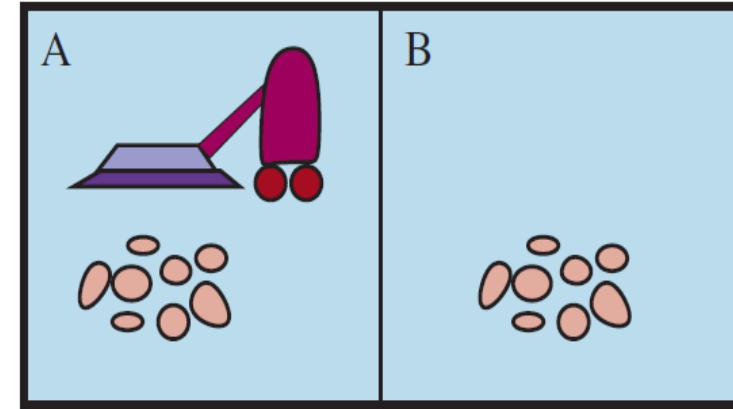
- Rasyonel ajan hareket tanımı:

- Verilen bilgiler doğrultusunda, performans ölçüsünün beklenen deęerini maksimize eden eylem



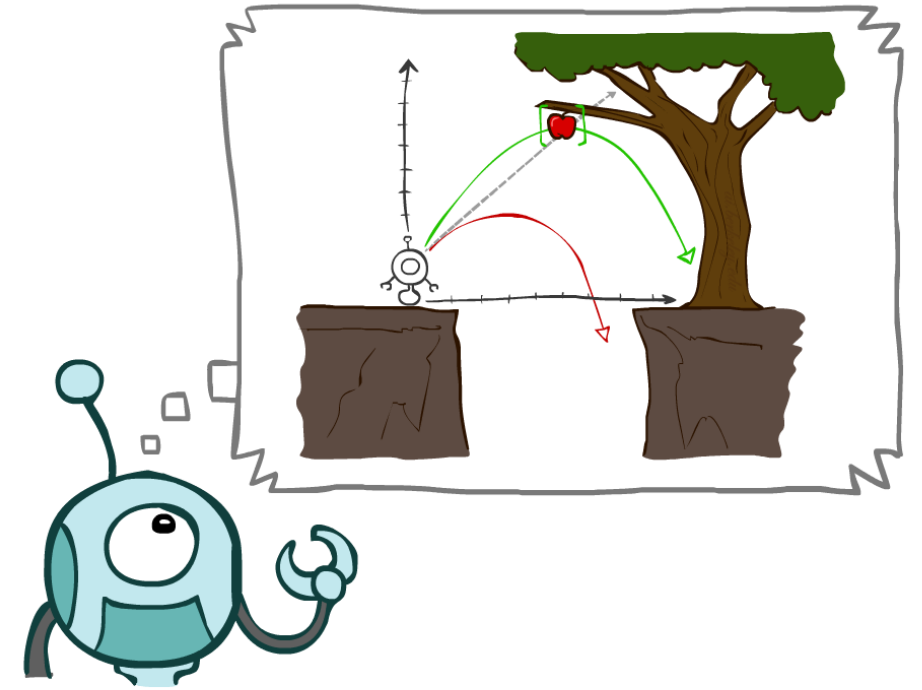
Elektrikli Süpürge Ajanı

- Performans Ölçütü
 - Varsayalım ki: ajan her adımda her temiz kare için bir puan ödüllendirilir.
- Çevre ile İlgili Ön Bilgi
 - Çevre için durum bilgisi bilinir *fakat* “kirli alanların dağılımı” ve ajanların başlangıç yerleri bilinmez.
- Ajanın gerçekleştirebileceği eylemler
 - Left, Right, Suck, NoOp
- Ajanın algı geçmişi
 - Ajan yerini ve yerin kir içerip içermediğini kolayca algılar



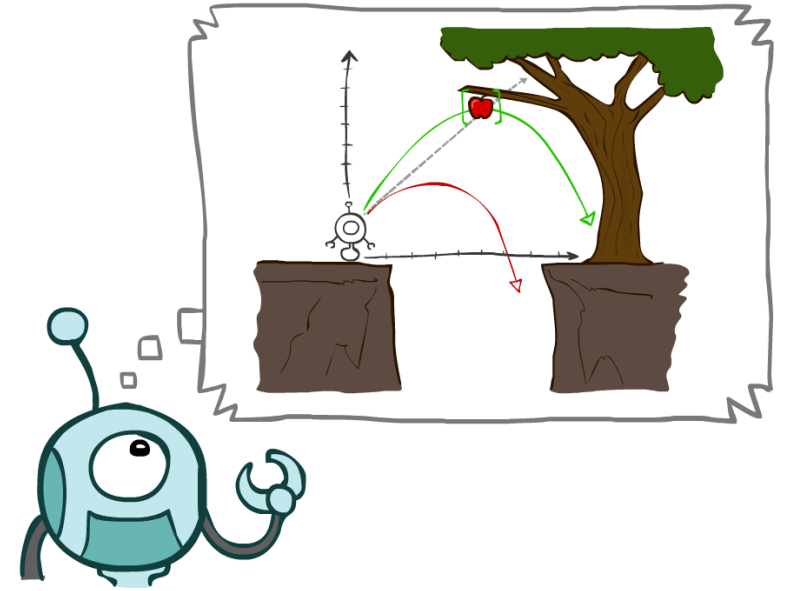
Herseyi Bilme, Öğrenme, Otonomluk

- Rasyonellik = Mükemmellik ?
- Rasyonel Hareket
 - Beklenen performansı maksimize eder.
- Mükemmel Hareket
 - Gerçek performansı maksimize eder.
- Rasyonel Ajan = Otonom Ajan



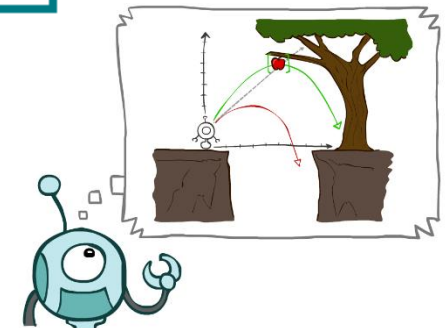
Görev Ortamının Doğası (Nature Environment)

- Görev ortamı 4 temel eleman içerir
 - **P**erformance Measure
 - **E**nvironment
 - **A**ctuators
 - **S**ensors



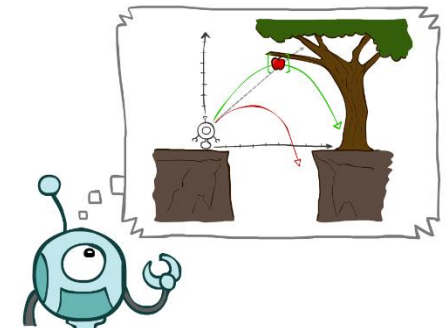
PEAS Örnek: Taksi Soförü

Agent Type	Performance Measure	Environment	Actuators	Sensors
Taxi driver	Safe, fast, legal, comfortable trip, maximize profits, minimize impact on other road users	Roads, other traffic, police, pedestrians, customers, weather	Steering, accelerator, brake, signal, horn, display, speech	Cameras, radar, speedometer, GPS, engine sensors, accelerometer, microphones, touchscreen



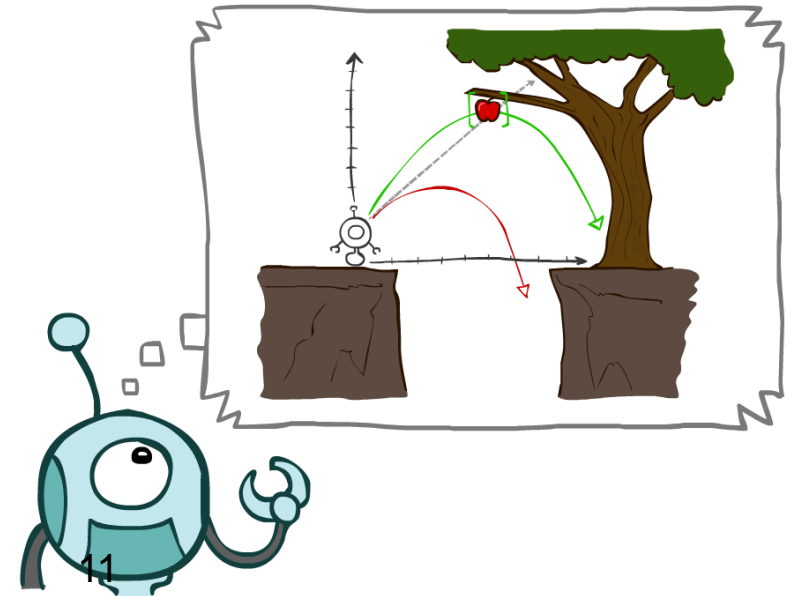
Örnekler: Ajanlar / PEAS Tanımlamaları

Agent Type	Performance Measure	Environment	Actuators	Sensors
Medical diagnosis system	Healthy patient, reduced costs	Patient, hospital, staff	Display of questions, tests, diagnoses, treatments	Touchscreen/voice entry of symptoms and findings
Satellite image analysis system	Correct categorization of objects, terrain	Orbiting satellite, downlink, weather	Display of scene categorization	High-resolution digital camera
Part-picking robot	Percentage of parts in correct bins	Conveyor belt with parts; bins	Jointed arm and hand	Camera, tactile and joint angle sensors
Refinery controller	Purity, yield, safety	Refinery, raw materials, operators	Valves, pumps, heaters, stirrers, displays	Temperature, pressure, flow, chemical sensors
Interactive English tutor	Student's score on test	Set of students, testing agency	Display of exercises, feedback, speech	Keyboard entry, voice



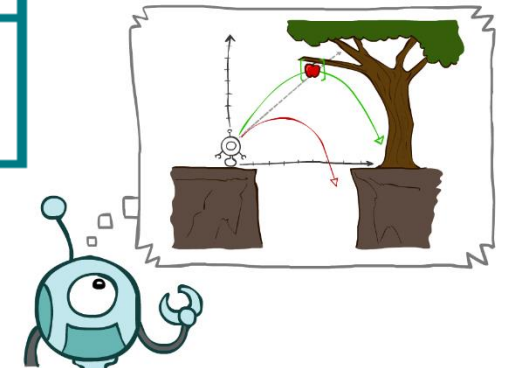
Görev Ortamının Özellikleri

- Fully observable vs. partially observable
 - Tamamen gözlemlenebilir ya da kısmen gözlemlenebilir
- Deterministic vs. stochastic / strategic
 - Deterministik ya da stokastik
- Episodic vs. sequential
 - Epizodik ya da sıralı
- Static vs. dynamic
 - Statik ya da dinamik
- Discrete vs. continuous
 - Kesikli ya da Sürekli
- Single agent vs. multiagent
 - Tek Ajan ya da Çoklu Ajan



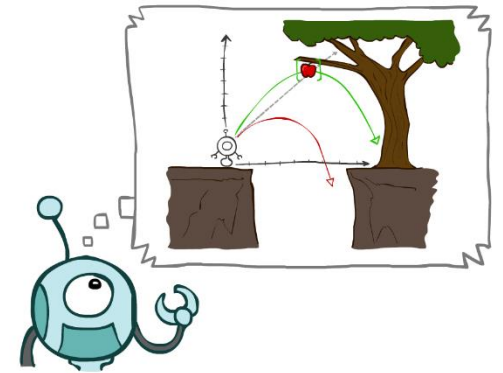
Görev Ortamının Örnekleri

Task Environment	Observable	Agents	Deterministic	Episodic	Static	Discrete
Crossword puzzle	Fully	Single	Deterministic	Sequential	Static	Discrete
Chess with a clock	Fully	Multi	Deterministic	Sequential	Semi	Discrete
Poker	Partially	Multi	Stochastic	Sequential	Static	Discrete
Backgammon	Fully	Multi	Stochastic	Sequential	Static	Discrete
Taxi driving	Partially	Multi	Stochastic	Sequential	Dynamic	Continuous
Medical diagnosis	Partially	Single	Stochastic	Sequential	Dynamic	Continuous
Image analysis	Fully	Single	Deterministic	Episodic	Semi	Continuous
Part-picking robot	Partially	Single	Stochastic	Episodic	Dynamic	Continuous
Refinery controller	Partially	Single	Stochastic	Sequential	Dynamic	Continuous
English tutor	Partially	Multi	Stochastic	Sequential	Dynamic	Discrete



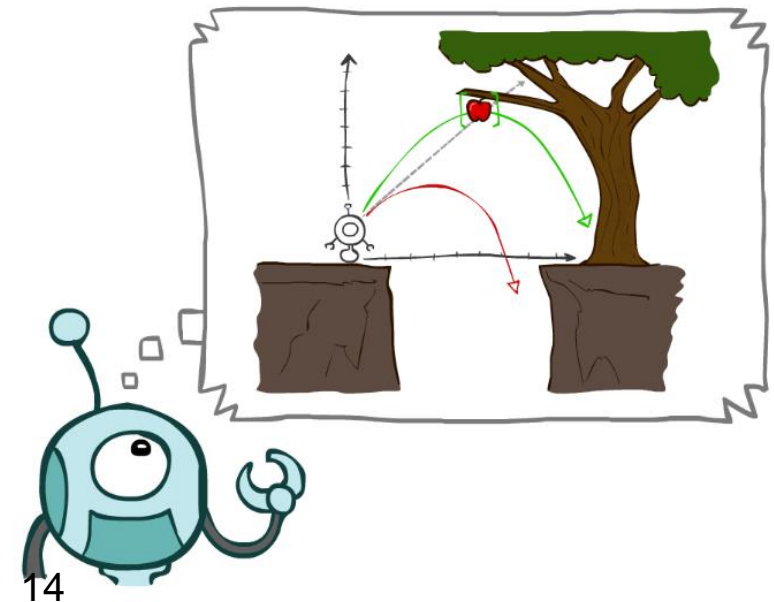
En Zorlu Durumlar

- Gerçek ortam özellikleri:
 - “Inaccessible”, “stochastic”, “dynamic”, “continuous”
- Ortamların olasılıkları çok fazladır:
 - “Accessible”, “deterministic”, “static”, “discrete”
- Sınırlı kaynaklar:
 - Bellek, zaman, yetersiz bilgi
- En zor ortam:
 - partially observable, multiagent, non deterministic, sequential, dynamic, continuous, unknown



Zeki Ajan Tasarımı

- **Yapay Zeka'nın Görevi:**
 - Ajan Programlar (Agent Program) tasarlamak
- **Ajan programlar mimari üzerinde çalışırlar**
 - Mimari = Algılayıcılar (Sensors) + Harekete Geçiriciler (Actuators)
- **Ajan:**
 - Ajan = Mimari + Program

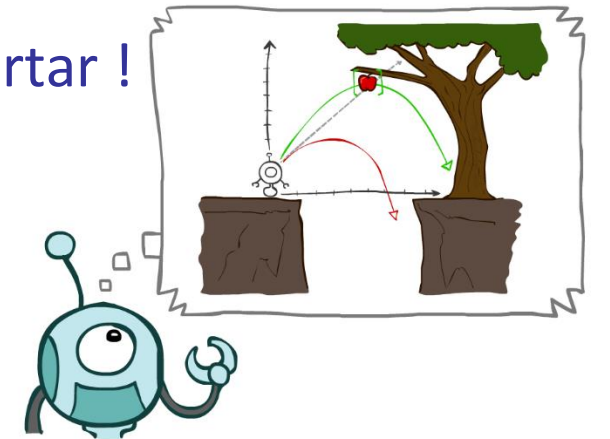


Ajan Türleri

- Ajan Türleri:

- Arama Tablosu Ajanları (Table Driven Agents)
- Basit Refleks Ajanları
- Model Tabanlı Refleks Ajanları
- Hedef Tabanlı Ajanlar
- Fayda Tabanlı Ajanlar

- Yukarıdan aşağıya doğru karmaşık ortamlarla başa çıkma becerisi artar !



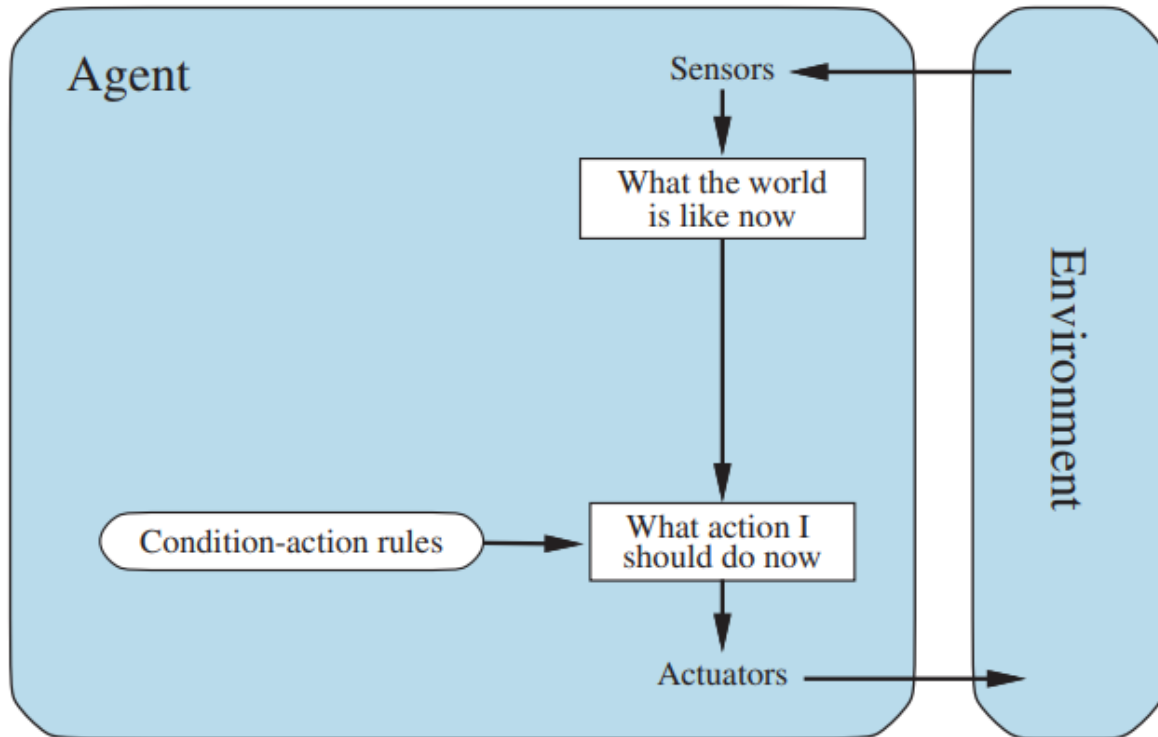
Arama Tablosu Ajani

function TABLE-DRIVEN-AGENT(*percept*) **returns** an action
 persistent: *percepts*, a sequence, initially empty
 table, a table of actions, indexed by percept sequences, initially fully specified

 append *percept* to the end of *percepts*
 action \leftarrow LOOKUP(*percepts*, *table*)
 return *action*

Figure 2.7 The TABLE-DRIVEN-AGENT program is invoked for each new percept and returns an action each time. It retains the complete percept sequence in memory.

Basit Refleks Ajanları



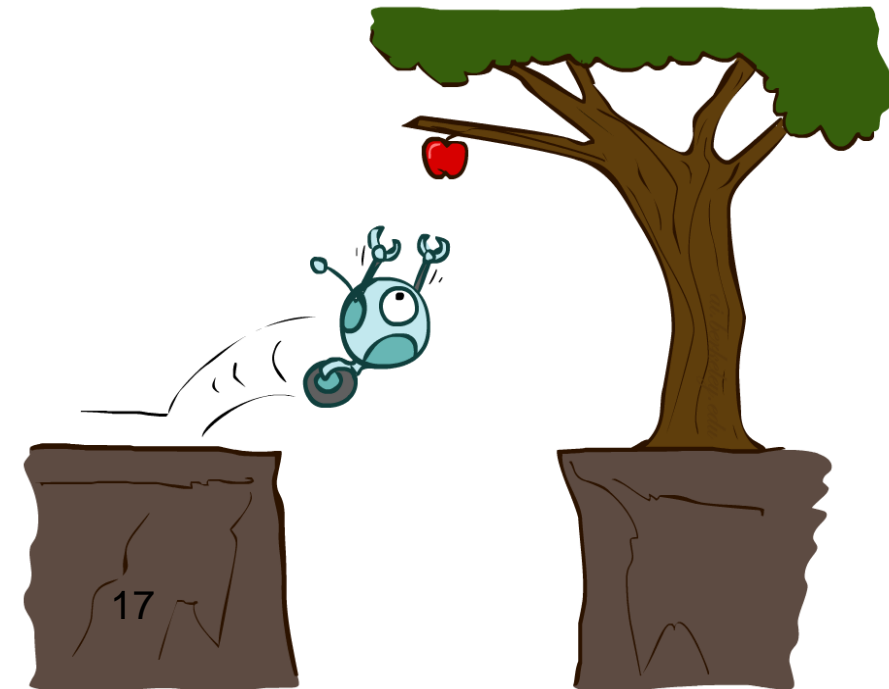
function SIMPLE-REFLEX-AGENT(*percept*) **returns** an action
persistent: *rules*, a set of condition–action rules

state \leftarrow INTERPRET-INPUT(*percept*)

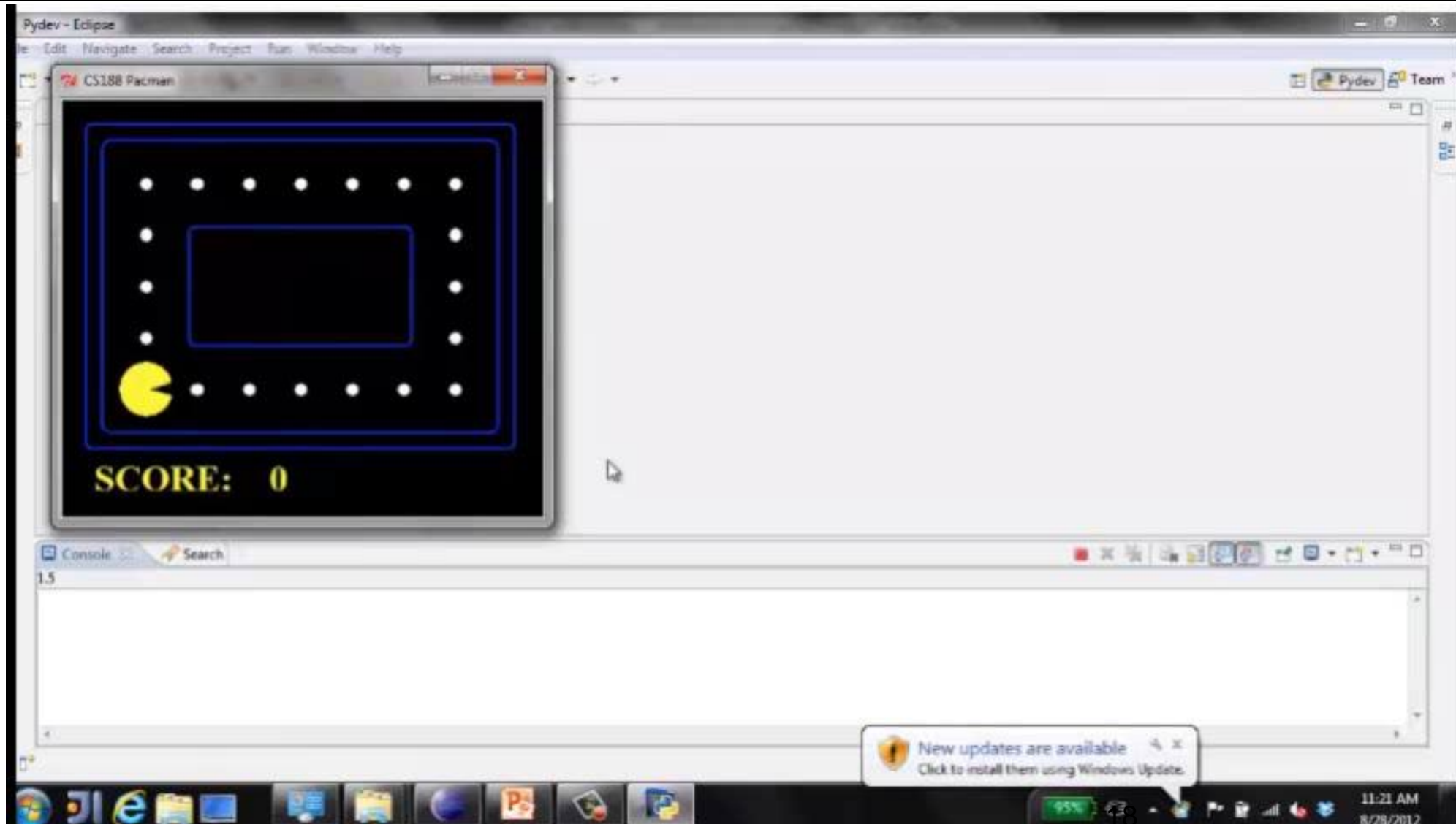
rule \leftarrow RULE-MATCH(*state*, *rules*)

action \leftarrow *rule*.ACTION

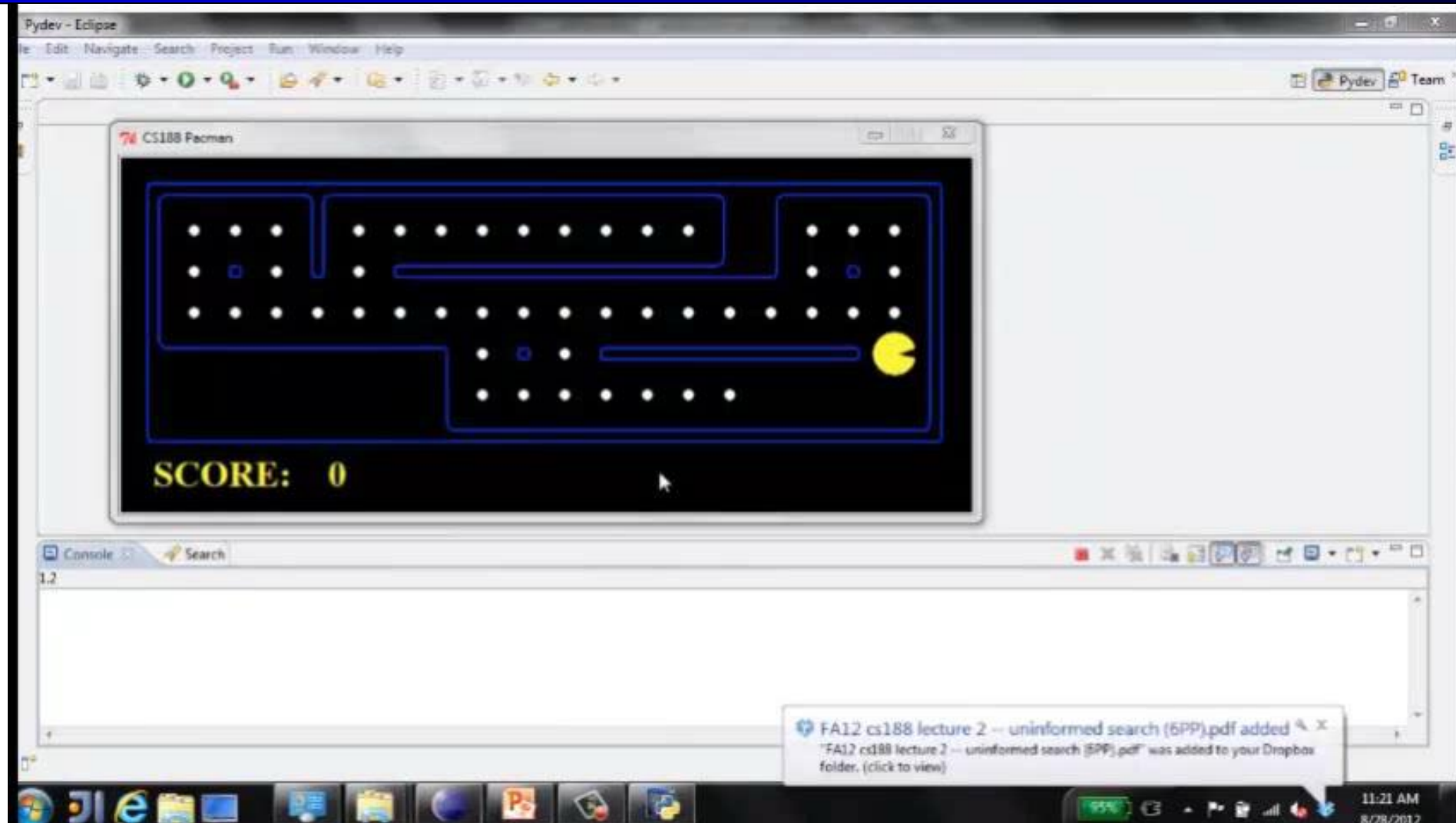
return *action*



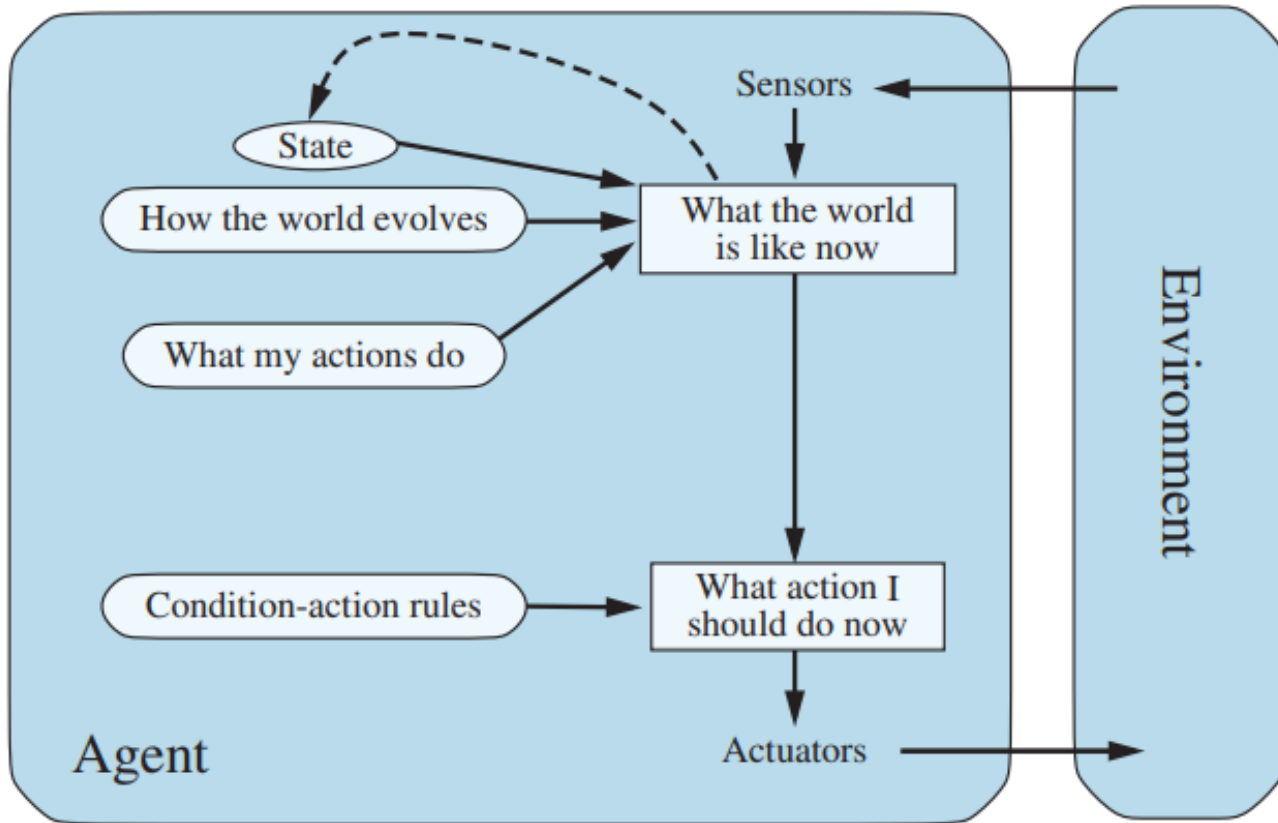
Refleks Tabanlı Ajan Optimal Durum Örneği



Refleks Tabanlı Ajan Başarısız Durum Örneği

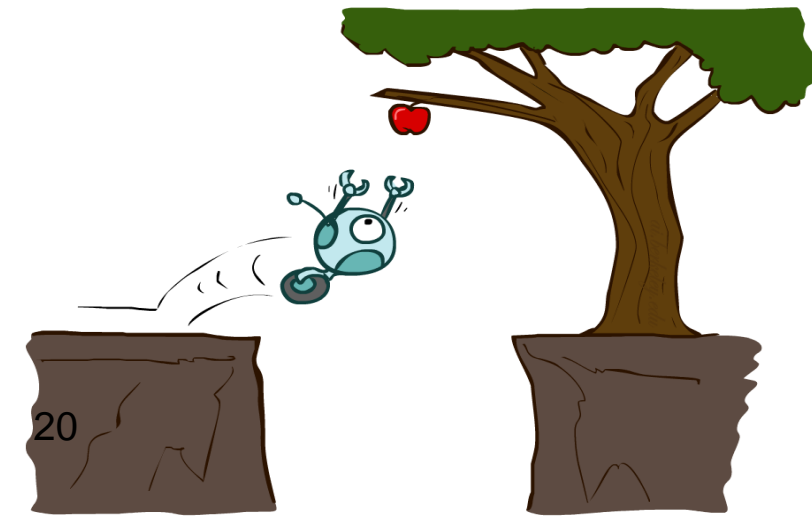


Model Tabanlı Refleks Ajanları

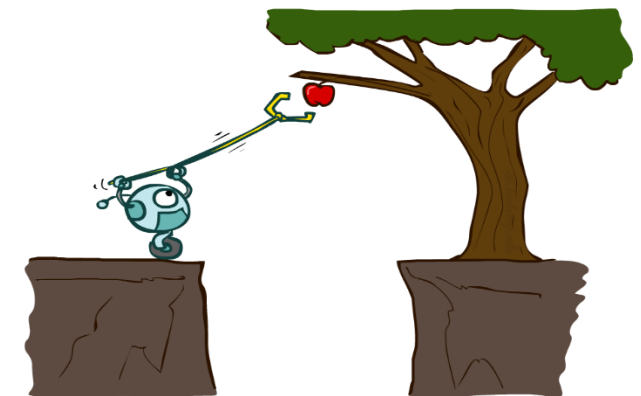
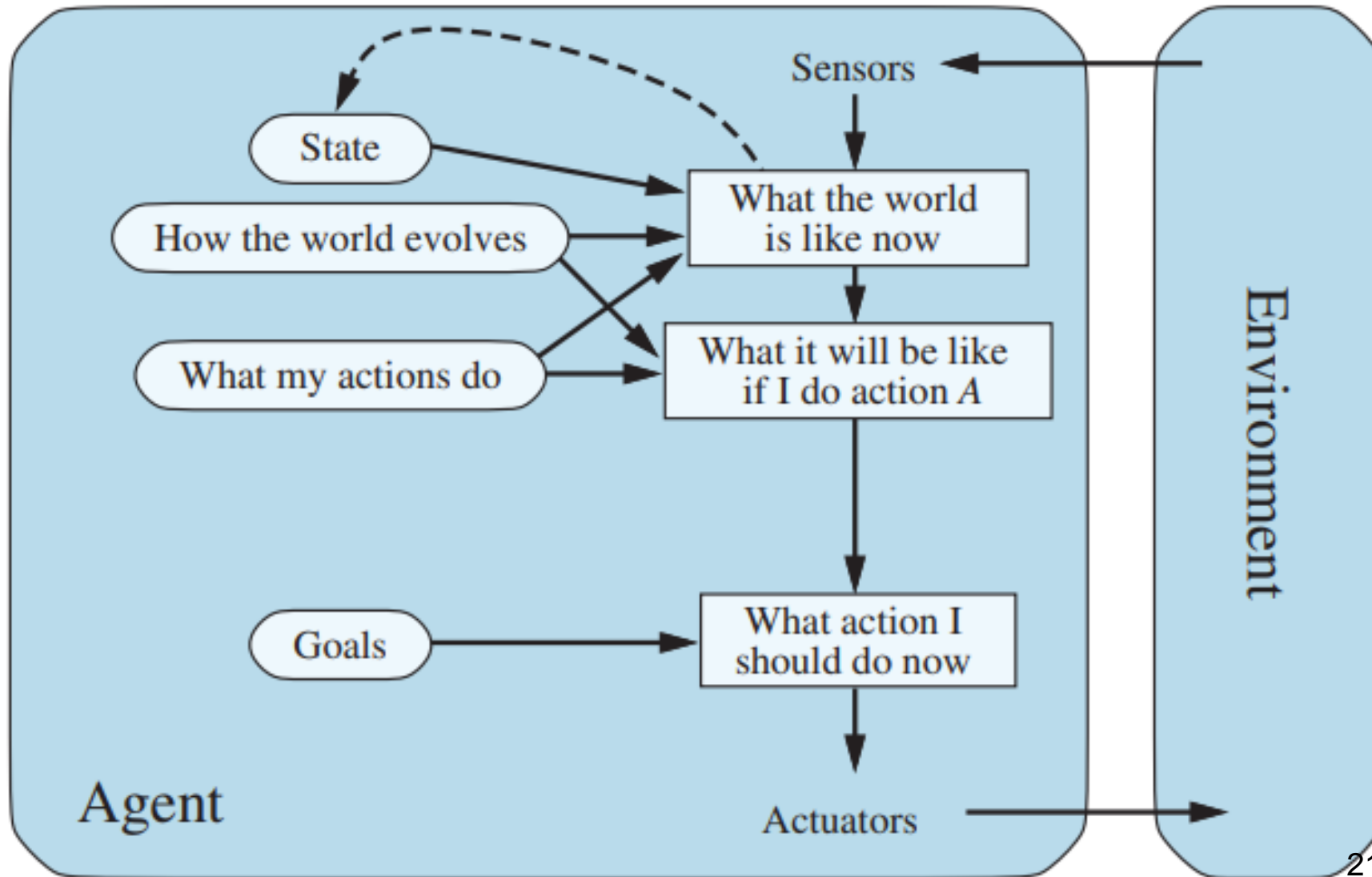


function MODEL-BASED-REFLEX-AGENT(*percept*) **returns** an action
persistent: *state*, the agent's current conception of the world state
transition_model, a description of how the next state depends on the current state and action
sensor_model, a description of how the current world state is reflected in the agent's percepts
rules, a set of condition-action rules
action, the most recent action, initially none

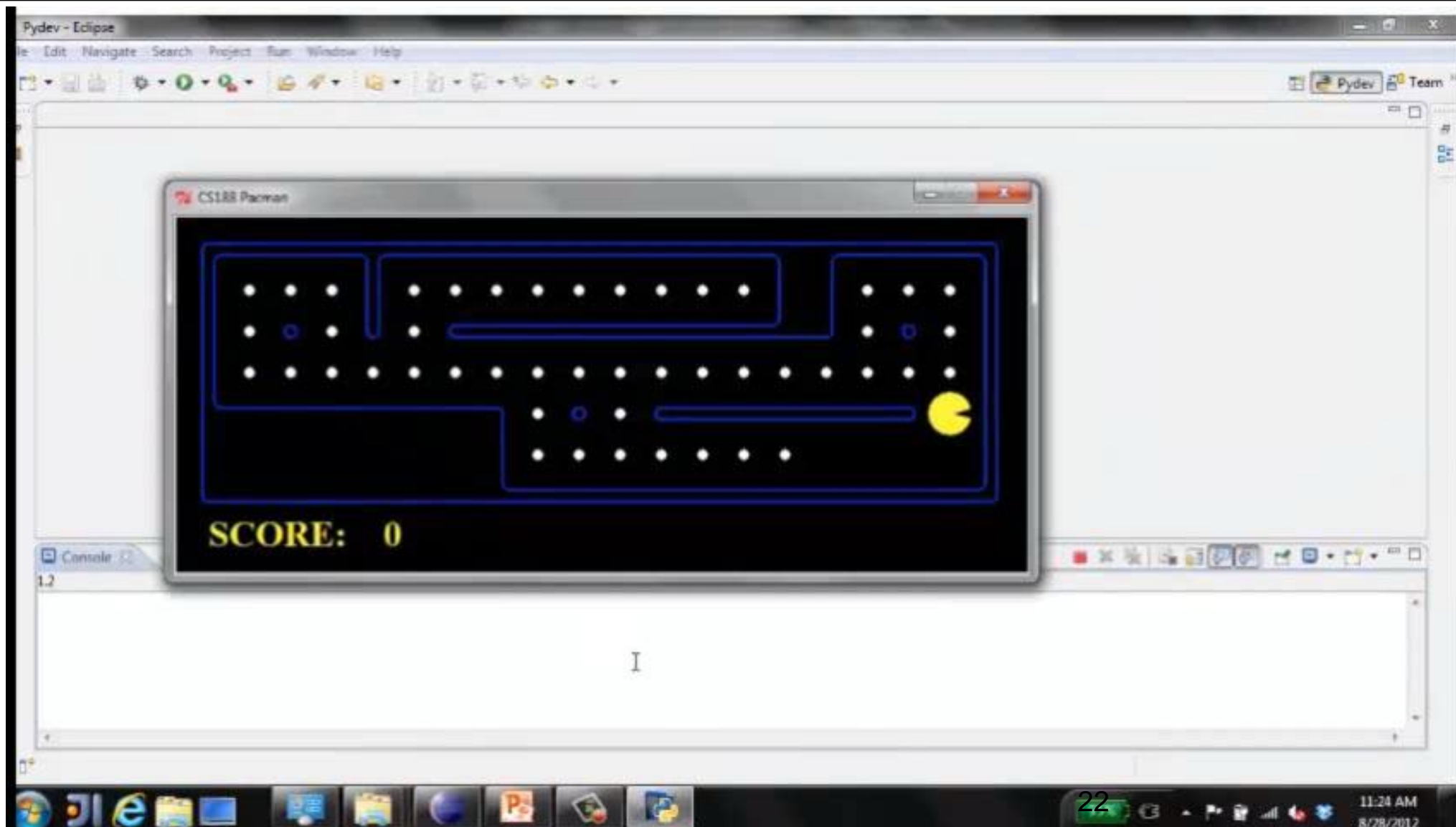
```
state ← UPDATE-STATE(state, action, percept, transition_model, sensor_model)
rule ← RULE-MATCH(state, rules)
action ← rule.ACTION
return action
```



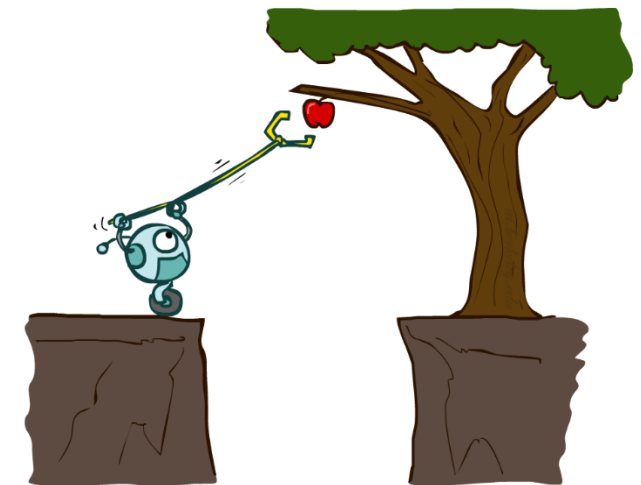
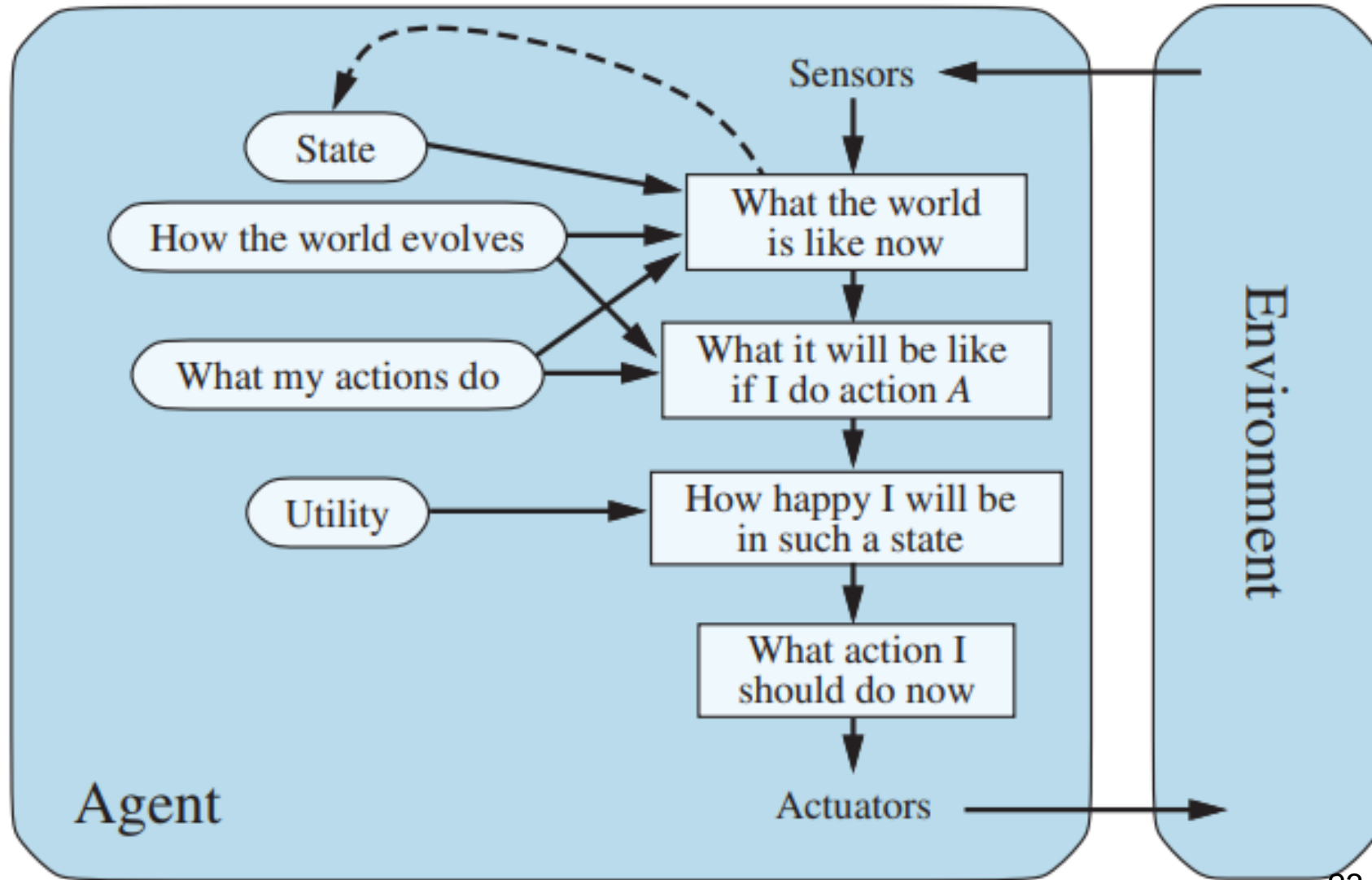
Hedef Tabanlı Ajanlar



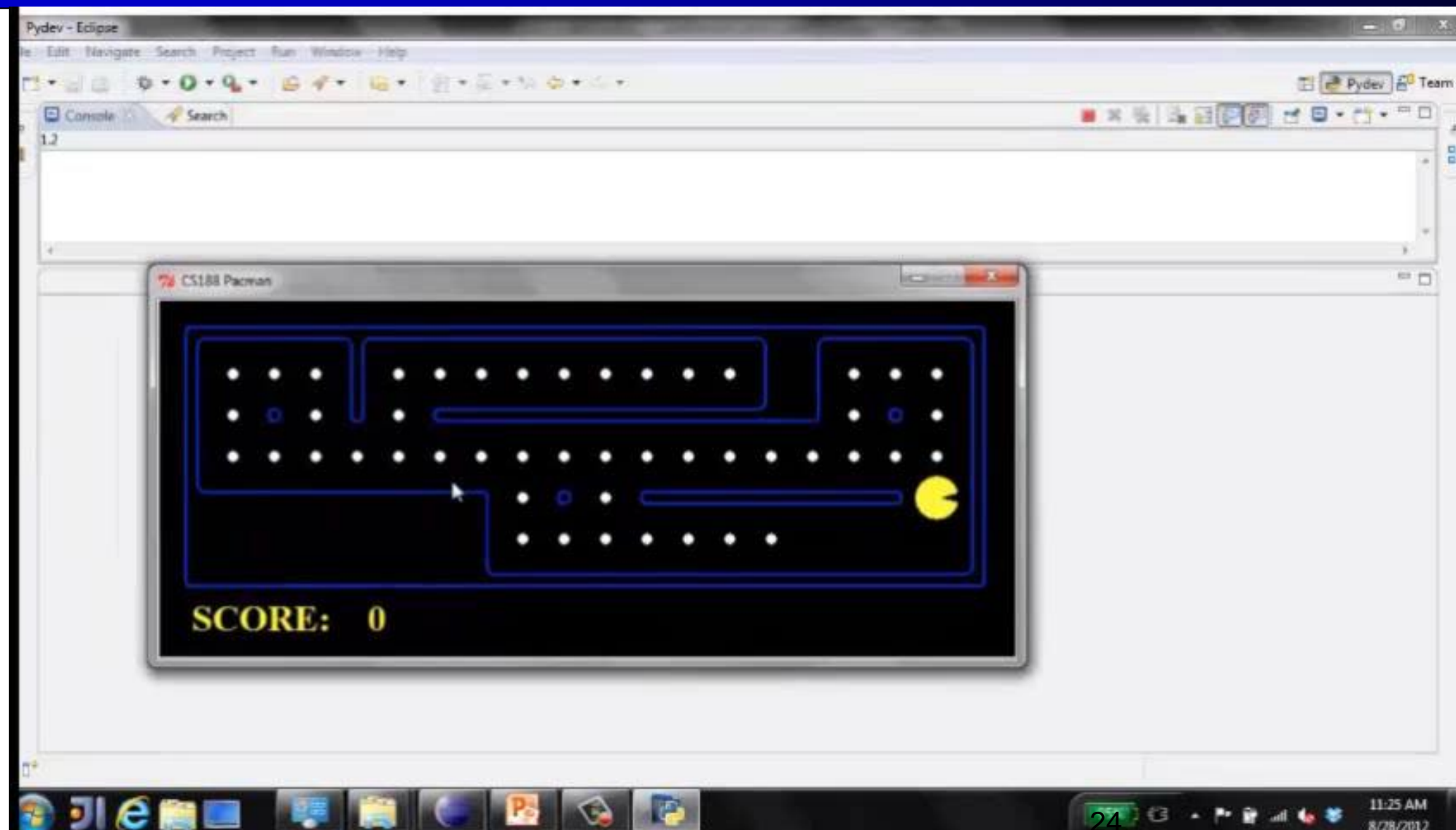
Video of Demo Replanning



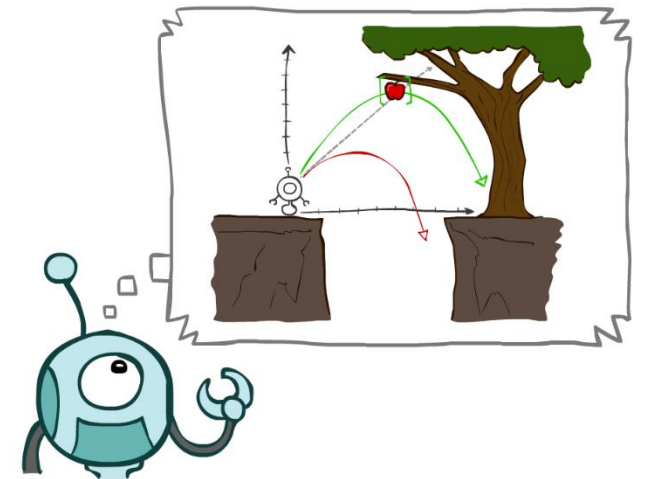
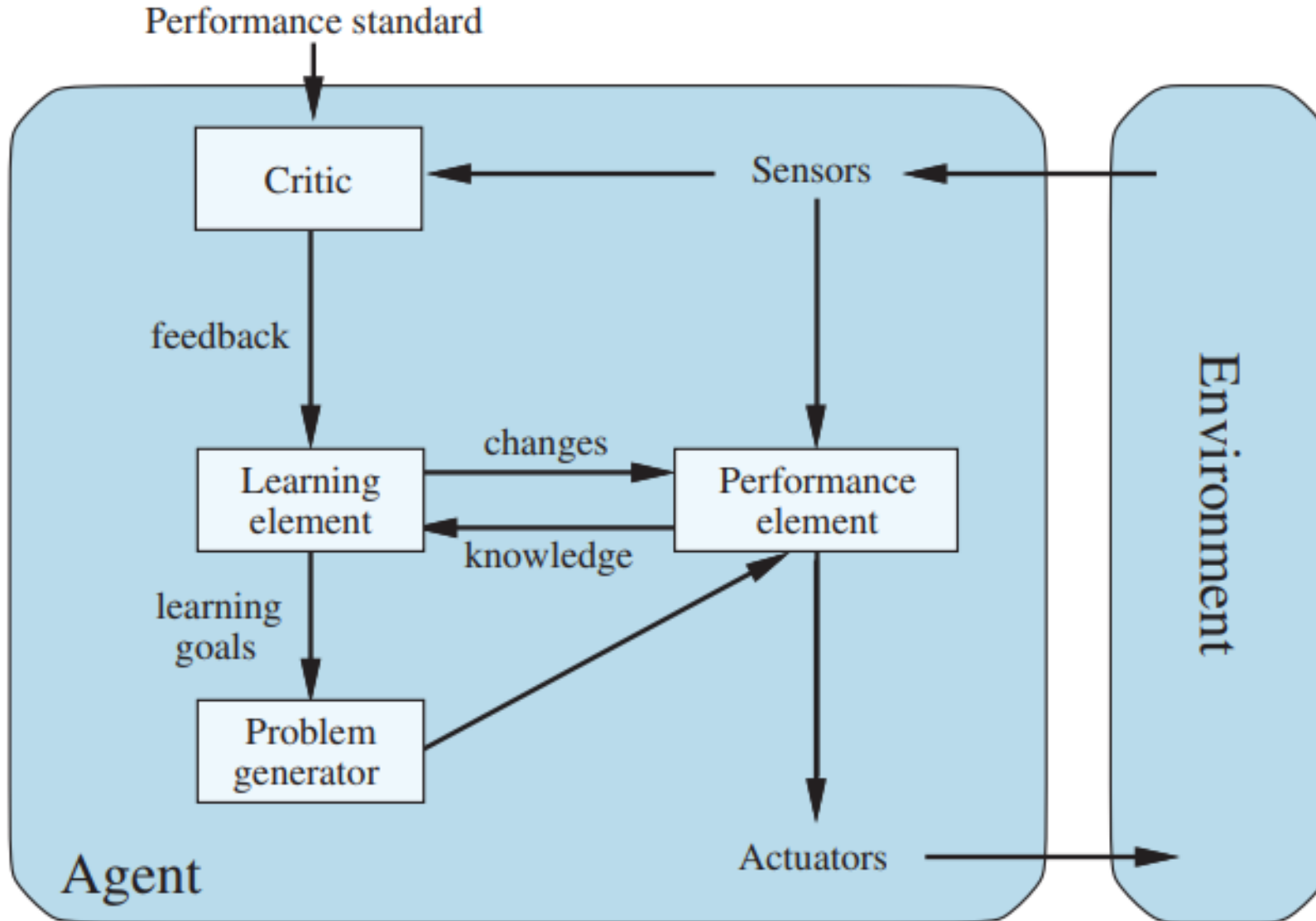
Fayda Tabanlı Ajanlar



Video of Demo Mastermind

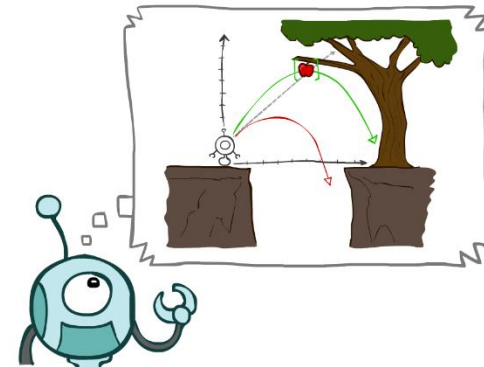


Öğrenme Ajanları



Özet

- Bir **ajan** bulunduğu çevreyi algılar ve harekete geçer, bir ajan program ile uygulanır.
- Bir **ideal ajan** daima beklenen performansı maksimum yapacak aksiyonları seçer.
- Bir **otonom ajan** tasarımcı tarafından belirlenen çevre özelliklerini kullanmaktansa kendi tecrübelerini kullanır.
- Bir **ajan program** algıdan (percept) →aksiyona (action) eşleştirme yapar ve kendi durumunu günceller.
 - **Reflex agents** oluşan algılara anında cevap verir.
 - **Goal-based agents** amaçlara ulaşmak için belirli sırada hareket eder.
 - **Utility-based** kendi utility function(fayda fonksiyonu) değerini maksimum yapar.
- **Bilgi temsili** başarılı bir ajan dizaynı için önemlidir.
- Bazı çevre (**environments**) seçenekleri diğerlerine göre ajanlar için daha zordur.
 - En zorlu çevreler inaccessible, nondeterministic, dynamic, ve continuous özellikli olanlardır.



Referanslar

- Artificial Intelligence A Modern Approach, Stuart Russell and Peter Norvig, Prentice Hall Series in Artificial Intelligence.
- <https://inst.eecs.berkeley.edu/~cs188/su20/> (Nikita Kitaev Ders Notları) 2020
- http://aytugonan.cbu.edu.tr/YZM3217/LectureNotes/YZM3217_lecture2.pdf
- <http://ce.sharif.edu/courses/91-92/2/ce417-1/resources/root/Lectures/Chapter-2.pdf>
- <https://katie.cs.mtech.edu/classes/archive/f15/csci446/slides/01%20-%20Agents-Statespaces.pdf>
- <https://www.youtube.com/watch?v=vRA4RUL5dvk>
- <https://www.javatpoint.com/types-of-ai-agents>
- [BİL551 – YAPAY ZEKA AJANLAR \(AGENTS\) - ppt indir \(slideplayer.biz.tr\)](http://bil551-yapay-zeka-ajanlar-agents-ppt-indir.slideplayer.biz.tr)
- [Hafta 3 ZEKİ ETMENLER \(INTELLIGENT AGENTS\) - ppt indir \(slideplayer.biz.tr\)](http://hafta-3-zeki-etmenler-intelligent-agents-ppt-indir.slideplayer.biz.tr)