

# انواع محیط

1. کاملاً مشاهده پذیر - مشاهده پذیر جزئی - غیر قابل مشاهده
2. قطعی - اتفاقی - راهبردی
3. اپیزودیک - غیر اپیزودیک
4. ایستا - پویا - نیمه پویا
5. گسسته - پیوسته
6. تک عاملی - چند عاملی

# ۱. انواع محیط (مشاهده پذیری)



□ **مشاهده پذیری** به میزان اطلاعاتی که عامل (توسط حسگرهایش) از محیط دریافت می کند بستگی دارد.

۱. **کاملاً مشاهده پذیر (Fully Observable):** عامل در هر لحظه می تواند به تمام اطلاعات مرتبط با محیط دسترسی داشته باشد.

▪ مثال: شطرنج (عامل می تواند کل وضعیت صفحه را ببیند).

۲. **مشاهده پذیر جزئی (Partially Observable):** عامل فقط به بخشی از اطلاعات محیط دسترسی دارد و ممکن است اطلاعاتی از برخی قسمت های محیط نداشته باشد.

▪ مثال: بازی پوکر (عامل فقط کارتهای خودش را می بیند و اطلاعاتی از کارتهای سایر بازیکنان ندارد).



۳. **غیر قابل مشاهده (Unobservable):** عامل هیچ اطلاعات مستقیمی از محیط دریافت نمی کند و باید بر اساس فرضیات یا تجربیات قبلی خود عمل کند.

▪ مثال: بازی در یک محیط تاریک بدون حسگرهای دیداری.



## ۲. انواع محیط (قطعیت)

□ **قطعیت**: یعنی عامل چقدر می‌تواند پیش‌بینی کند که انجام یک کار، چه نتیجه‌ای خواهد داشت.

۱. **محیط قطعی (Deterministic)**: خروجی یک عمل همیشه مشخص است و هیچ عدم قطعیتی وجود ندارد.  
▪ مثال: شطرنج (هر حرکت اثر مشخصی دارد و تصادفی نیست).



۲. **محیط تصادفی (Stochastic)**: نتایج اعمال عامل تحت تأثیر تصادف و عدم قطعیت هستند.  
▪ مثال: تاس انداختن در بازی مار و پله.



۳. **محیط راهبردی (Strategic)**: علاوه بر تصمیمات عامل، تصمیمات دیگر بازیگران (عوامل هوشمند دیگر) بر محیط تأثیر می‌گذارند.

▪ مثال: پوکر (بازیگران دیگر نیز حرکت می‌کنند و تصمیمات آن‌ها بر بازی تأثیر دارد).



### ۳. انواع محیط (اپیزودیک بودن)

□ **اپیزودیک بودن** یعنی این که هر کاری که عامل انجام می‌دهد، روی کارهای بعدی تأثیر دارد یا نه.

۱. **محیط اپیزودیک (Episodic):** هر عملی که عامل انجام می‌دهد، روی کارهای بعدی تأثیر ندارد و هر اقدام نتیجه جداگانه‌ای دارد

مثال: کنترل کیفی در کارخانه (تصمیم برای هر محصول مستقل از محصولات قبلی است).

۲. **محیط غیر اپیزودیک-ترتیبی (Sequential):** اعمال عامل بر تصمیمات آینده (کارهای بعدی) تأثیر دارد و وابستگی زمانی وجود دارد.

مثال: شطرنج (حرکات گذشته بر وضعیت آینده بازی تأثیر دارند).



## ۴. انواع محیط (پویایی)

□ **پویایی محیط** نشان می‌دهد که آیا محیط خودش تغییر می‌کند یا فقط وقتی که عامل کاری انجام می‌دهد

۱. **محیط ایستا (Static):** محیط در طول زمان تغییر نمی‌کند، مگر زمانی که عامل عملی انجام دهد.  
مثال: شطرنج (اگر بازیکنی حرکت نکند، صفحه بازی ثابت می‌ماند).

۲. **محیط پویا (Dynamic):** محیط بدون مداخله عامل تغییر می‌کند.  
مثال: بازی Super Mario Bros (وضعیت محیط در طول زمان تغییر می‌کند)

۳. **محیط نیمه پویا (Semi-Dynamic):** محیط ثابت است، اما عامل با گذشت زمان جریمه می‌شود.  
مثال: بازی شطرنج با محدودیت زمانی (محیط تغییر نمی‌کند، اما زمان برای تصمیم‌گیری محدود است).



## ۵. انواع محیط (گسستگی)

□ **گسستگی محیط** یعنی ببینیم که تغییرات در محیط مرحله به مرحله (گسسته) است یا به صورت پیوسته و روان انجام می شود..



۱. **محیط گسسته (Discrete):** تعداد حالات و اعمال ممکن محدود و قابل شمارش است. (با یک متغیر گسسته action ها نمایش داده می شوند)  
مثال: شطرنج (حرکات محدود و مشخص هستند).



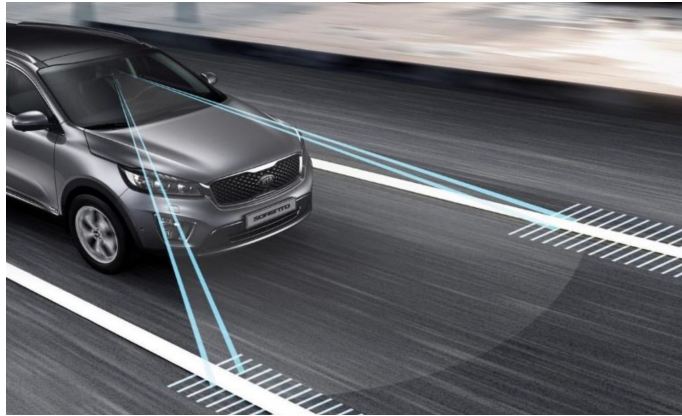
۲. **محیط پیوسته (Continuous):** حالات و اعمال ممکن بی نهایت و پیوسته هستند.  
مثال: رانندگی (موقعیت خودرو و فرمان به صورت پیوسته تغییر می کنند).



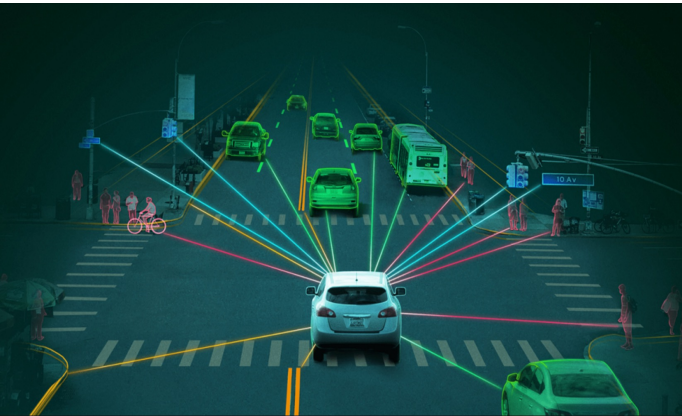
## ۶. انواع محیط (تعداد عامل‌ها)

□ **تعداد عامل‌ها** تعیین می‌کند که چه تعداد عامل هوشمند در محیط فعالیت می‌کنند.

۱. **محیط تک‌عاملی (Single-Agent Environment):** محیطی که فقط یک عامل هوشمند در آن فعالیت می‌کند و تعامل خاصی با دیگر عوامل ندارد.  
مثال: یک خودروی خودرانی که تنها عامل هوشمند در جاده است



۲. **محیط چندعاملی (Multi-Agent Environment):** محیطی که در آن چندین عامل هوشمند به‌طور هم‌زمان فعالیت دارند و تصمیمات آن‌ها بر یکدیگر تأثیر می‌گذارد.  
مثال: تعدادی خودروی خودرانی که در خیابان در حرکت هستند



# انواع محیط

محیط	قابل مشاهده	تعداد عامل‌ها	قطعی یا تصادفی	اپیزودیک یا ترتیبی	ایستا یا پویا یا نیمه‌پویا	گسسته یا پیوسته
جدول کلمات متقاطع	کاملاً قابل مشاهده	تک‌عاملی	قطعی	ترتیبی	ایستا	گسسته
شطرنج با ساعت	کاملاً قابل مشاهده	چندعاملی	قطعی	ترتیبی	نیمه‌پویا	گسسته
منچ	کاملاً قابل مشاهده	چندعاملی	تصادفی	ترتیبی	ایستا	گسسته
رانندگی تاکسی	نیمه قابل مشاهده	چندعاملی	تصادفی	ترتیبی	پویا	پیوسته
ربات انتخاب قطعات	نیمه قابل مشاهده	تک‌عاملی	تصادفی	اپیزودیک	پویا	پیوسته





# انواع عامل‌ها در مسائل جستجو

عامل‌های واکنشی (Reflex agents)

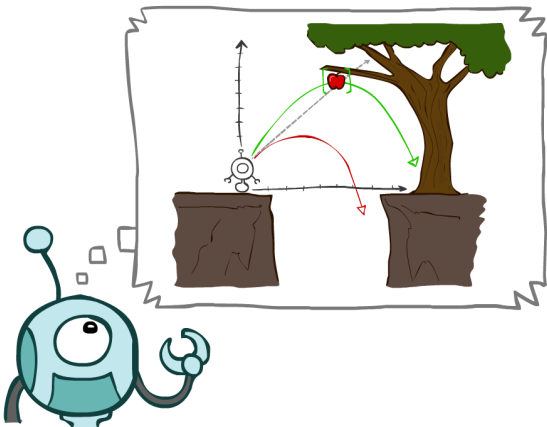
عامل‌های واکنشی ساده (Simple Reflex agents)

عامل‌های واکنشی مبتنی بر مدل (Model-based Agents)

عامل‌های برنامه‌ریز (Planning Agents)

عامل‌های هدف‌گرا (Goal-based Agents)

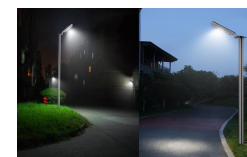
عامل‌های سودمندگرا (Utility-Based Agents)



# عامل‌های واکنشی (Reflex agents)

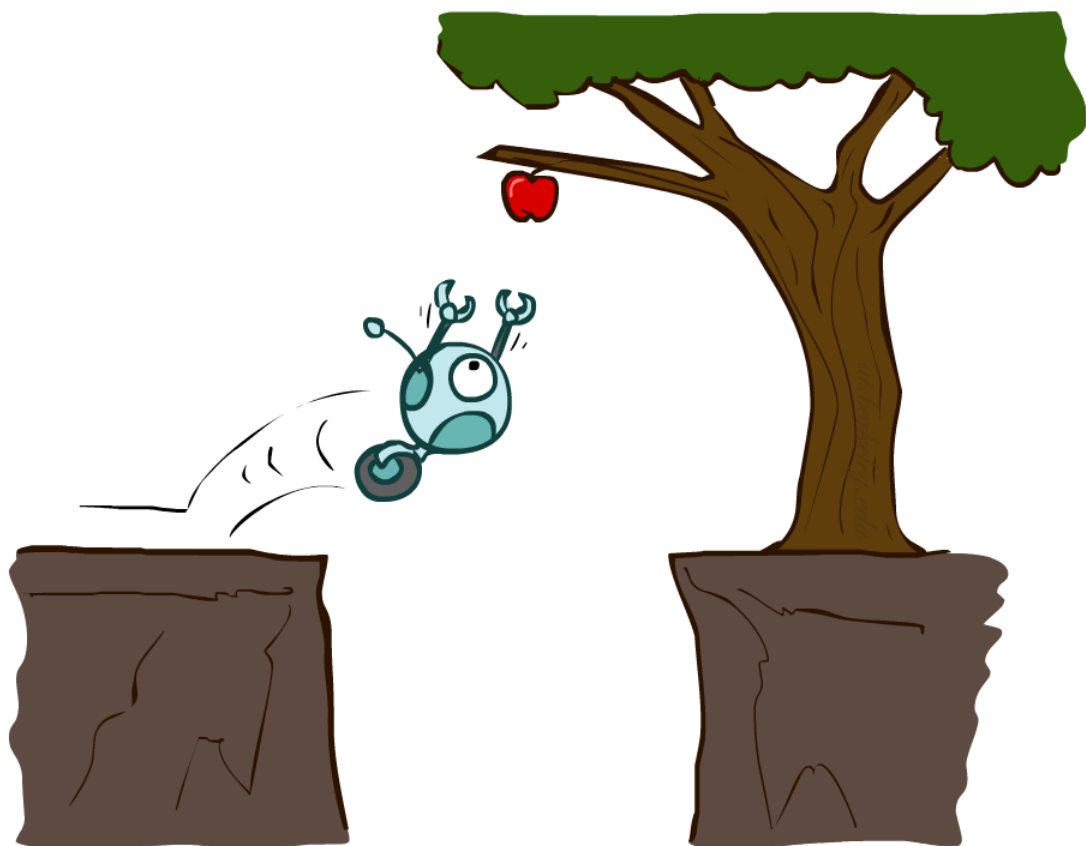
## □ عامل‌های واکنشی ساده (Simple Reflex agents)

- مقایسه عمل فعلی با لیستی از قوانین از پیش تعریف شده
  - قوانین به یک لیست گسترده‌ی "اگر... آنگاه..." شباهت دارند
- اجرای قانونی که با عمل فعلی تطابق دارد
- پیامدهای آینده‌ی عمل انتخاب شده خود را در نظر نمی‌گیرند
- تنها حالت فعلی محیط برایش اهمیت دارد
- مثال: حسگر نور در چراغ خیابان



## □ عامل‌های واکنشی مبتنی بر مدل (Model-based Agents):

- نسخه پیشرفته‌تری از عامل واکنشی ساده که مدل داخلی از محیط را در خود نگه میدارد
- مثال: جارورباتیک:
  - نقشه داخلی از اتاق ایجاد کرده

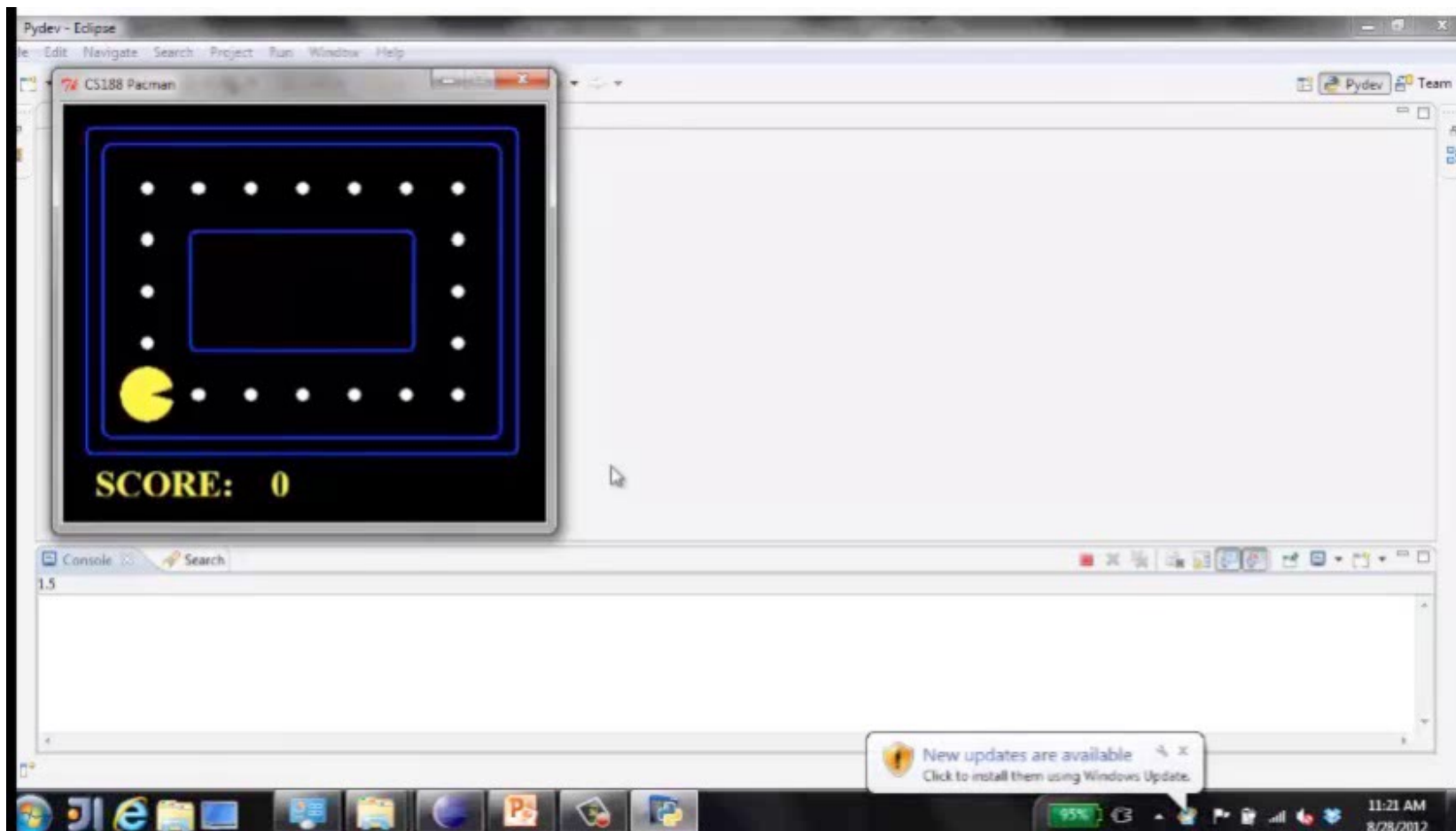


# عامل‌های واکنشی (Reflex agents)

اگر در خانه مجاور نقطه بود

آنگاه آن را بخور

✓ عامل واکنشی موفق می‌شود

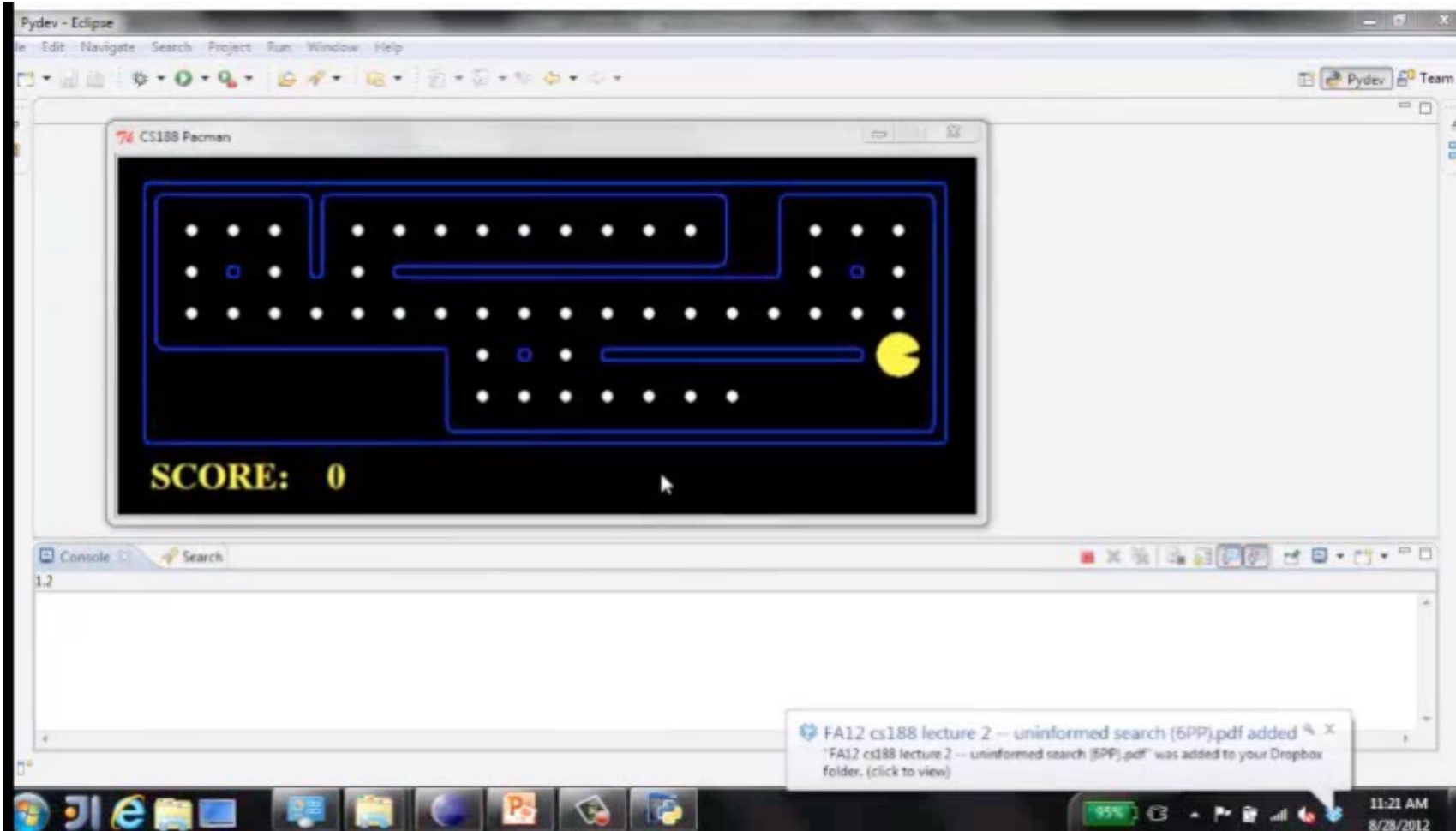


# عامل‌های واکنشی (Reflex agents)

اگر در خانه مجاور نقطه بود

آنگاه آن را بخور

✓ عامل واکنشی شکست می‌خورد



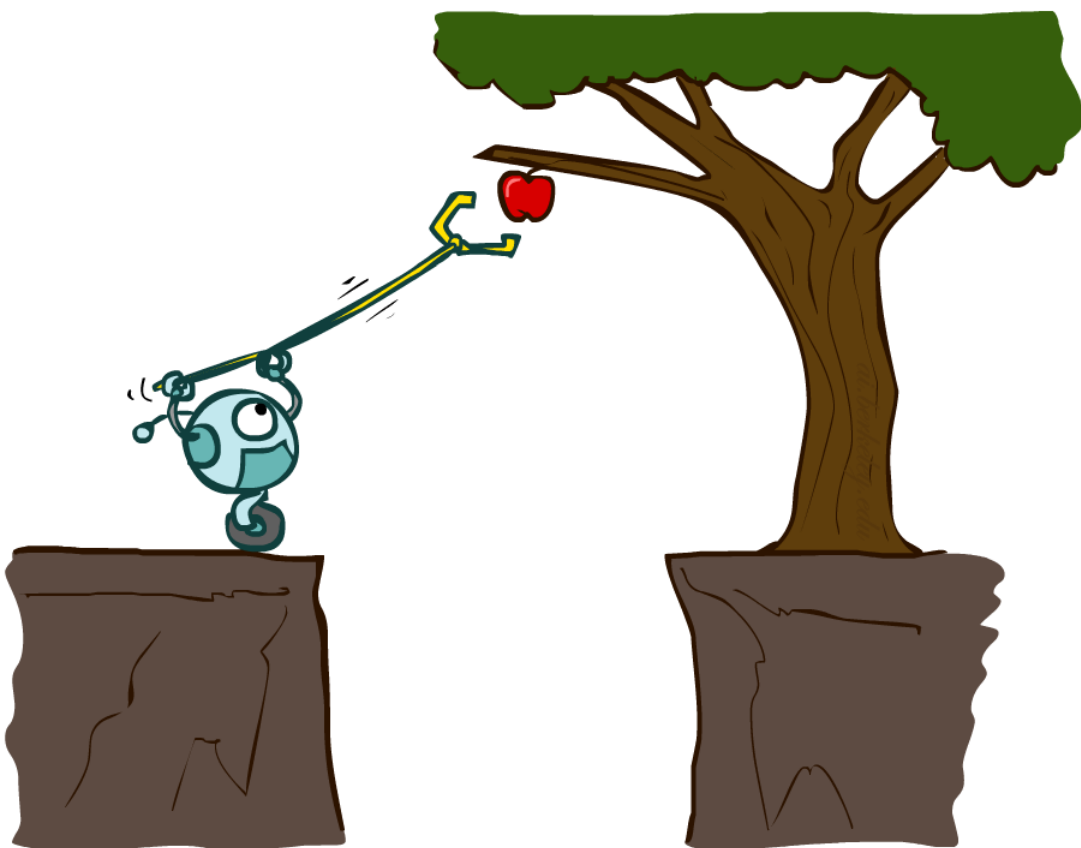
# عامل‌های برنامه‌ریز (Planning Agents)

## □ عامل‌های هدف‌گرا (Goal-based Agents):

- عامل دارای هدف است و برای رسیدن به هدف برنامه‌ریزی می‌کند
- می‌پرسد [چه می‌شود اگر این عمل را انجام دهم]
- در واقع عمل خود را ارزیابی می‌کند
  - بررسی درخت احتمالات:
  - هر شاخه نمایانگر یک عمل بالقوه است.
  - تحلیل پیامدهای هر عمل (محیط چگونه تغییر می‌کند-عواقب انجام آن عمل)
  - انتخاب عملی که بیشترین نزدیکی به هدف را دارد.
- پیامدهای آینده‌ی عمل انتخاب شده خود را در نظر می‌گیرند
- این عامل در صورت وجود راه حل حتماً به هدف می‌رسد (برنامه‌ریزی کامل)

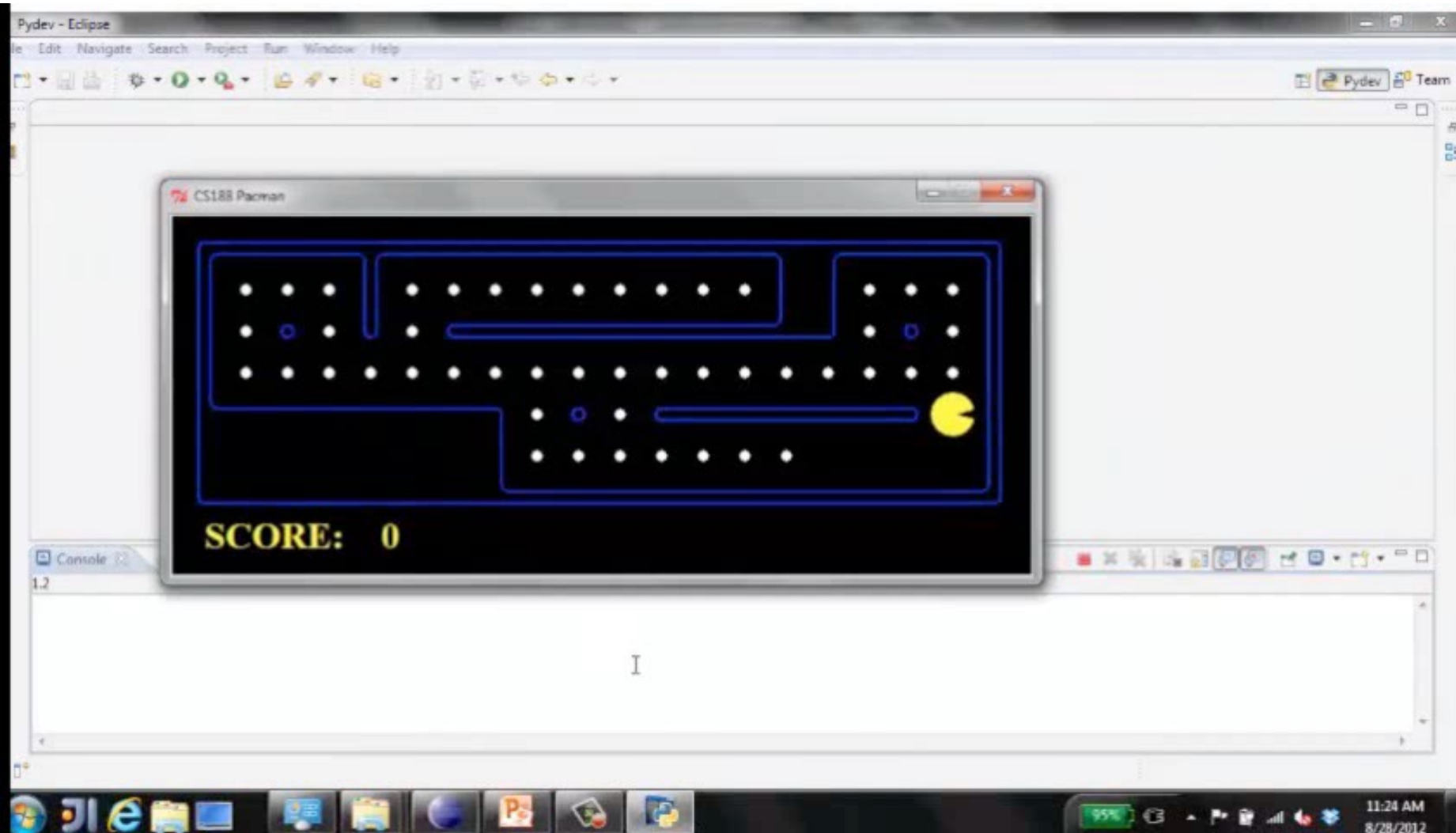
## □ عامل‌های سودمندگرا (Utility-Based Agents):

- نسخه پیشرفته‌تری از عامل‌های هدف‌گرا
- عاملی که با کمترین هزینه ممکن به هدف می‌رسد
- دارای برنامه‌ریزی بهینه در مقابل برنامه‌ریزی کامل
  - این عامل همواره بهترین راه حل ممکن را پیدا می‌کند با تا به هدف برسد (برنامه‌ریزی بهینه)



# عامل‌های برنامه‌ریز - هدف‌گرا

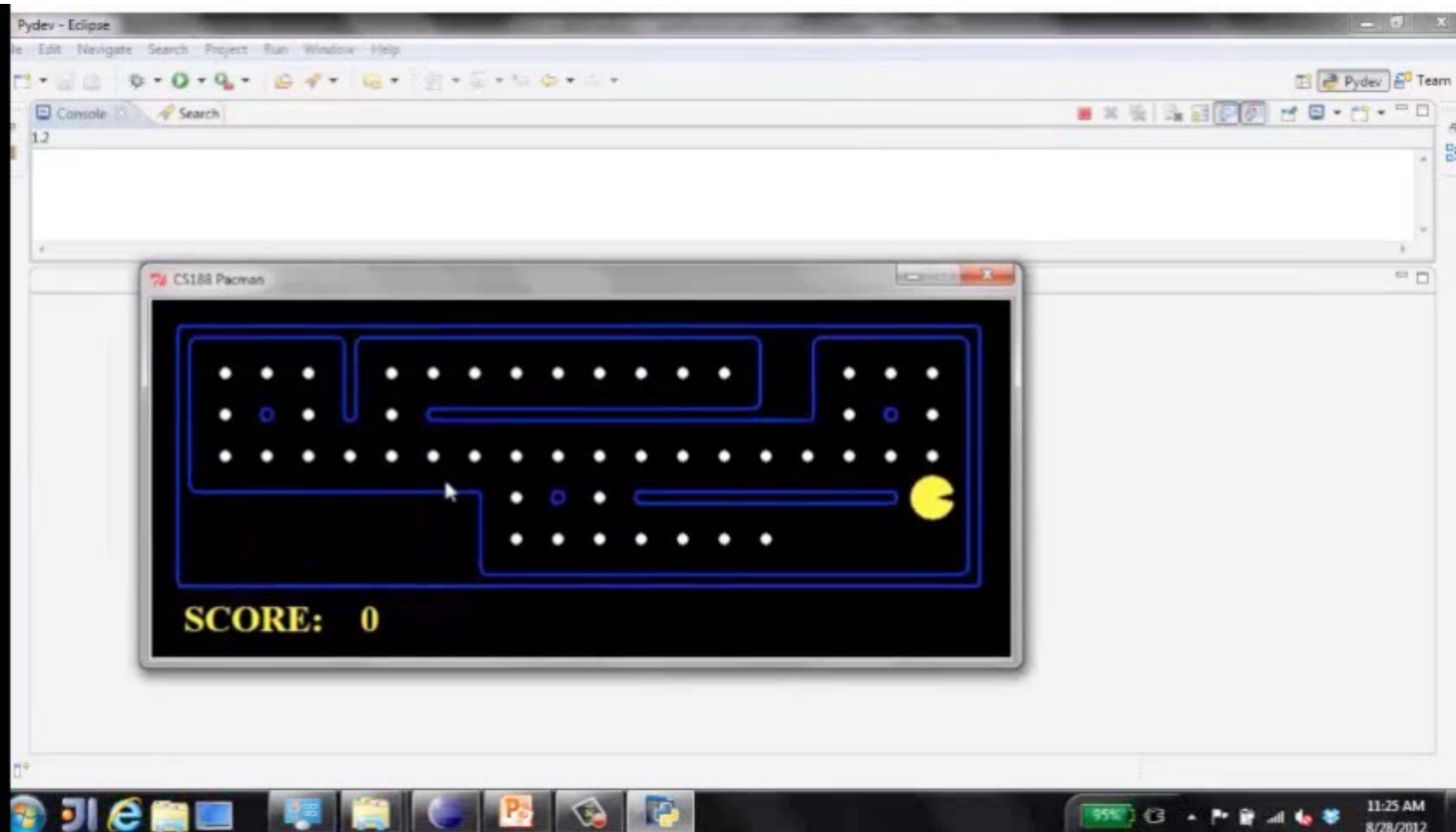
**هدف:** تمام نقاط را بخور





# عوامل های برنامه ریز - سودمند گرا

**هدف:** تمام نقاط را بخور  
(با کمترین عملیات ممکن)



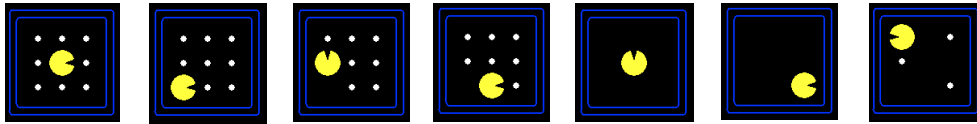
# مسائل جستجو



# مسائل جستجو

- یک مسئله جستجو شامل:
  - شامل فضای حالت، حالت اولیه، هدف و تابع جانشین است.

(یک فضای حالت) A state space



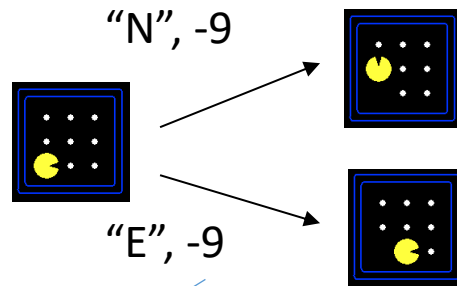
- فضای حالت (State Space):
  - همه‌ی وضعیت‌های ممکن

- حالت اولیه (Initial State):
  - نقطه‌ی شروع

- آزمون هدف (Goal Test):
  - مشخص می‌کند که آیا به هدف رسیده‌ایم یا نه
  - مثال:  $s$  هایی که هیچ نقطه‌ای در آن نباشد  $G(s) = \text{false}$

- تابع جانشین (Successor Function):
  - مشخص می‌کند که از یک وضعیت مشخص، چه اعمالی قابل انجام هستند، این اعمال ما را به چه وضعیت‌هایی می‌برند و هزینه‌ی هر انتقال چقدر است.

- یک راه‌حل، دنباله‌ای از عملیات (یک برنامه) است که حالت اولیه را به حالت هدف تبدیل می‌کند.

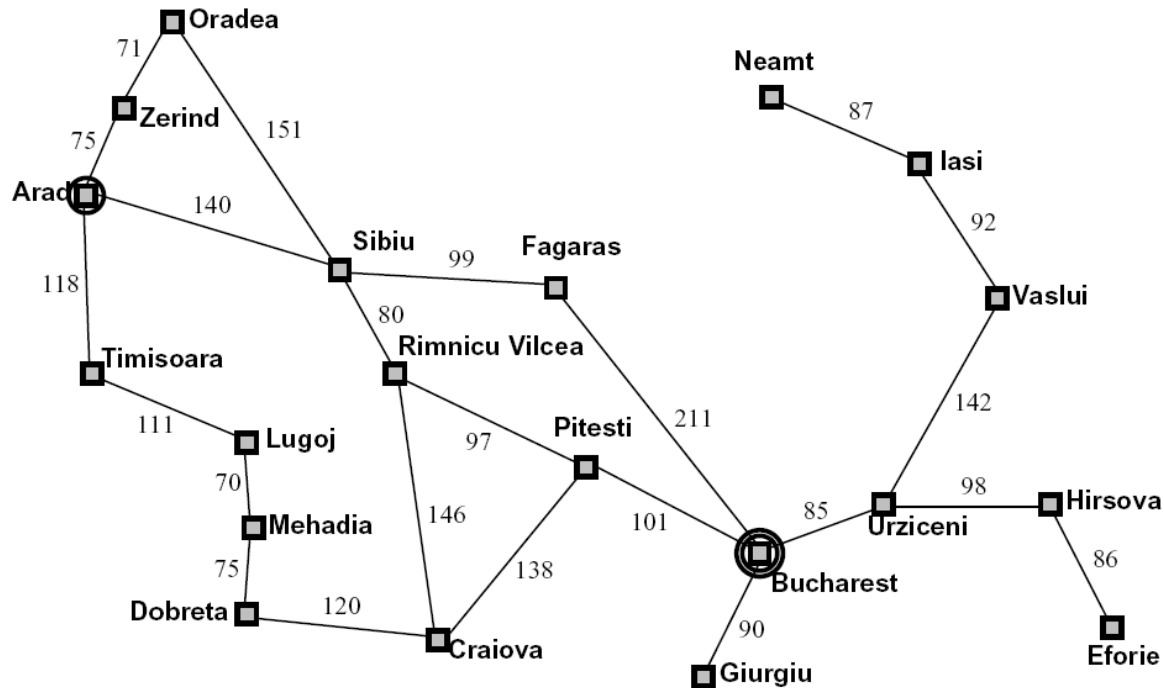


1+ برای هر قدم؛ 10- برای غذا؛

$$\text{Successor}(s) = \{(s', a, c) \mid a \in A(s), s' = \text{Result}(s, a), c = \text{Cost}(s, a, s')\}$$

- $s$ : وضعیت فعلی
- $A(s)$ : مجموعه‌ی اعمال ممکن در وضعیت  $s$
- $a$ : یک عمل از مجموعه‌ی  $A(s)$
- $s'$ : وضعیت جدید پس از اجرای  $a$ ، که از طریق تابع گذار  $\text{Result}(s, a)$  تعیین می‌شود
- $c(s, a, s')$ : هزینه‌ی انجام عمل  $a$  برای رسیدن از  $s$  به  $s'$

# مثال: مسیریابی در رومانی



□ فضای حالت (State Space):  
▪ شهرها

□ حالت اولیه (initial state):  
▪ شهر Arad

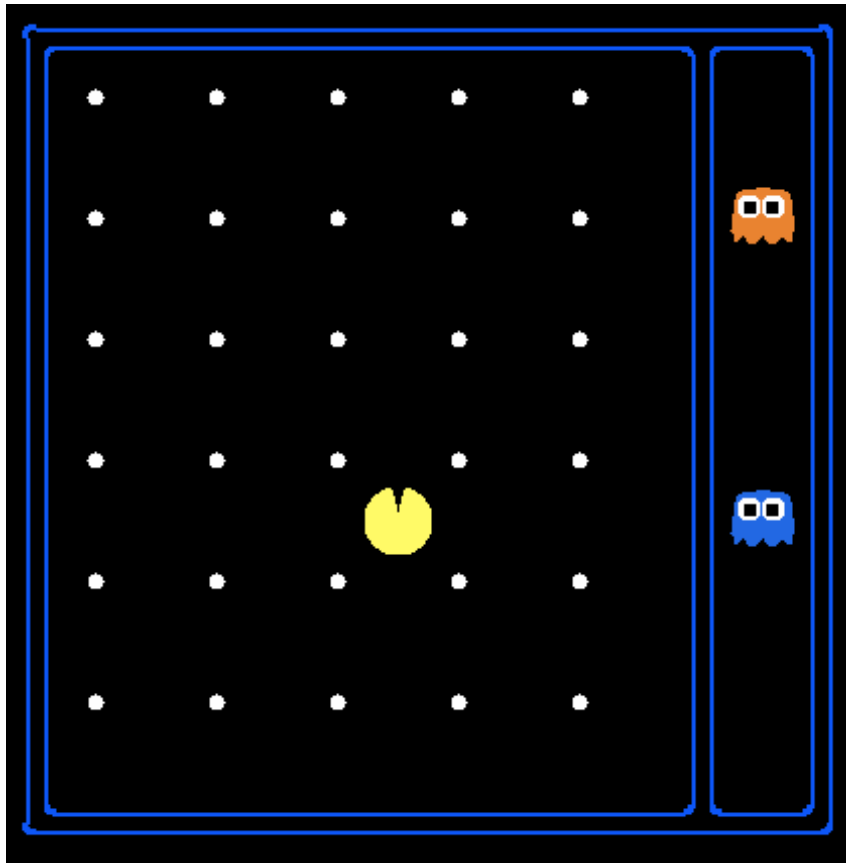
□ آزمون هدف (goal state):  
▪ آیا حالت == Bucharest؟

□ تابع جانشین (Successor Function):  
▪ عمل‌ها (Actions)  
• رانندگی به شهرهای مجاور

▪ هزینه‌ها (Costs)  
• هزینه = فاصله

□ راه‌حل، دنباله‌ای از شهرها، ...، Arad, Sibiu, Fagaras,

# اندازه فضای حالت



## □ فضای حالت:

- مکان‌های ممکن برای عامل: ۱۲۰
- جهت‌های ممکن برای عامل: ۴
- حالت‌های ممکن برای هر کدام از ارواح: ۱۲
- تعداد نقاط: ۳۰

## □ اندازه فضای جستجو

$$120 \times (2^{30}) \times (12^2) \times 4$$

- تعداد حالت‌ها اگر مسئله مسیریابی باشد؟  
120
- تعداد حالت‌ها اگر مسئله خوردن تمام نقاط باشد؟  
 $120 \times (2^{30})$