

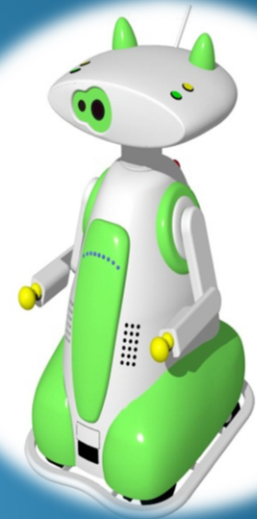
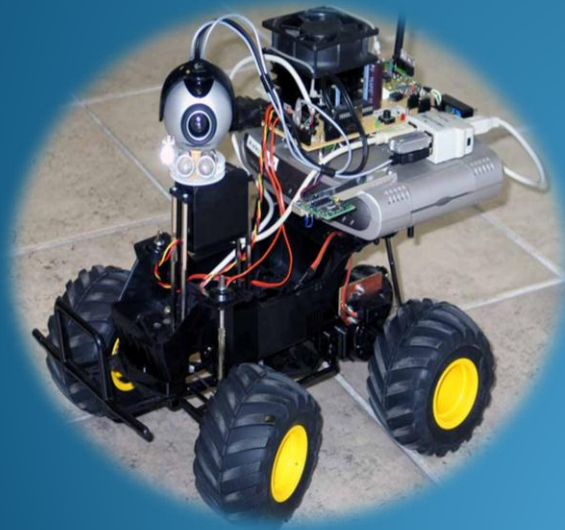


# هوش مصنوعی

مدرس : احمد ابدالی

# فصل دوم

## عوامل های هوشمند



# فصل دوم

عامل هوشمند چیست ؟

عامل های هوشمند چگونه عمل می کنند ؟

ساختار عامل های هوشمند

برنامه های عامل هوشمند

انواع عامل های هوشمند

انواع محیط ها و ...

# عامل هوشمند

عامل هوشمند : چیزی که محیط را درک و اقدام می کند

به عبارت دیگر :

عامل (agent) ابتدا از طریق حسگرها ( sensor ) اطلاعات محیط را دریافت می کند و سپس از طریق محرک ها ( actuator ) روی محیط عمل انجام می دهد.

تذکر : به محرک اثر گذار نیز می گویند ( effector )

تعامل با محیط = حسگرها + محرک ها

یک مثال در مورد عامل انسانی

حسگرها : چشم و گوش و ...

عامل : یک عامل انسانی

محرک ها : دست و پا و ...

# یک نمونه عامل هوشمند

کاوش گر مریخ :

چشم الکترونیک به  
عنوان حسگر

آنتن دریافت امواج رادیویی از  
زمین به عنوان حسگر

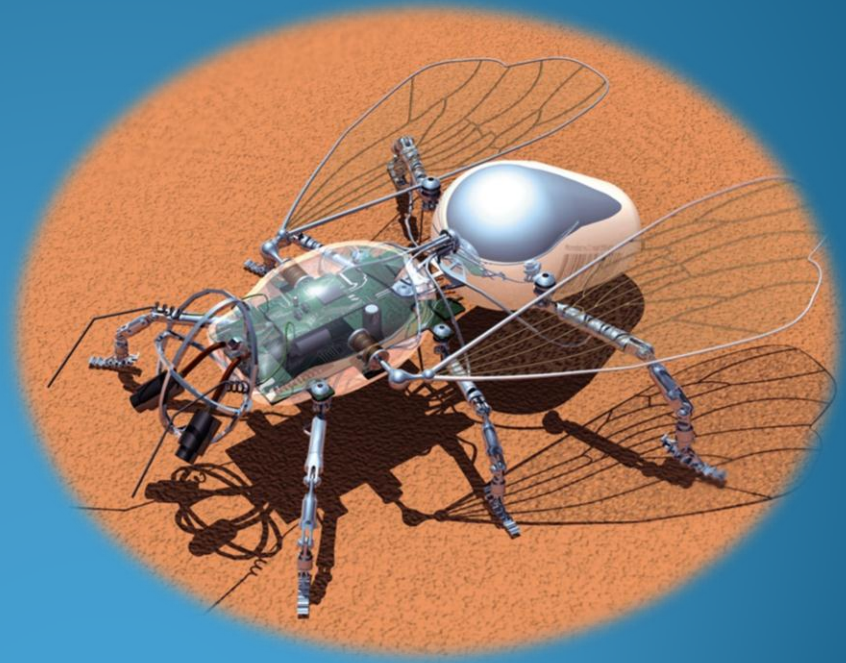
چرخ ها به عنوان  
محرك

عمل این عامل هوشمند حرکت کردن و اکتشاف در سطح مریخ و ارسال آنها به زمین است



# بحث کنید :

در عامل های هوشمند زیر حسگر ها و محرک ها و نوع عمل عامل را تعیین کنید



# یک عامل خوب چگونه عمل می کند ؟

یک عامل خوب عاملی است که بهترین کار را روی محیط انجام دهد . ( تعریف گنگ )

یک عامل خوب عاملی است کار درست انجام می دهد . ( تعریف گنگ )

یک عامل خوب عاملی است که موفق تر است . ( تعریف گنگ )

برای اینکه میزان موفقیت یک عامل را تعریف کنیم ابتدا معیار کارآیی را تعریف می کنیم .

توجه : مشخص است معیار کارآیی برای عامل های مختلف متفاوت است مثلاً معیار کارآیی برای عامل جارو برقی تمیز بودن زمین است . در حالی این معیار برای عامل کاوشگر مریخ نمونه برداری از سطح مریخ است .

به تعریف کلی زیر قانع می شویم :

معیار کارآیی از سوی ناظر خارجی تعیین می شود و کارایی عامل مورد نظر را تعیین می کنند .

# نتیجه بحث معیار کارآیی یک عامل

معیار کارآیی  $\approx$  درجه موفقیت یک عامل

معیار کارآیی  $\approx$  پارامتری برای مقایسه عامل های همسان

حداکثر کردن معیار کارایی مورد انتظار  $\approx$  عامل منطقی

حداکثر کردن معیار کارایی واقعی  $\approx$  عامل ایده آل



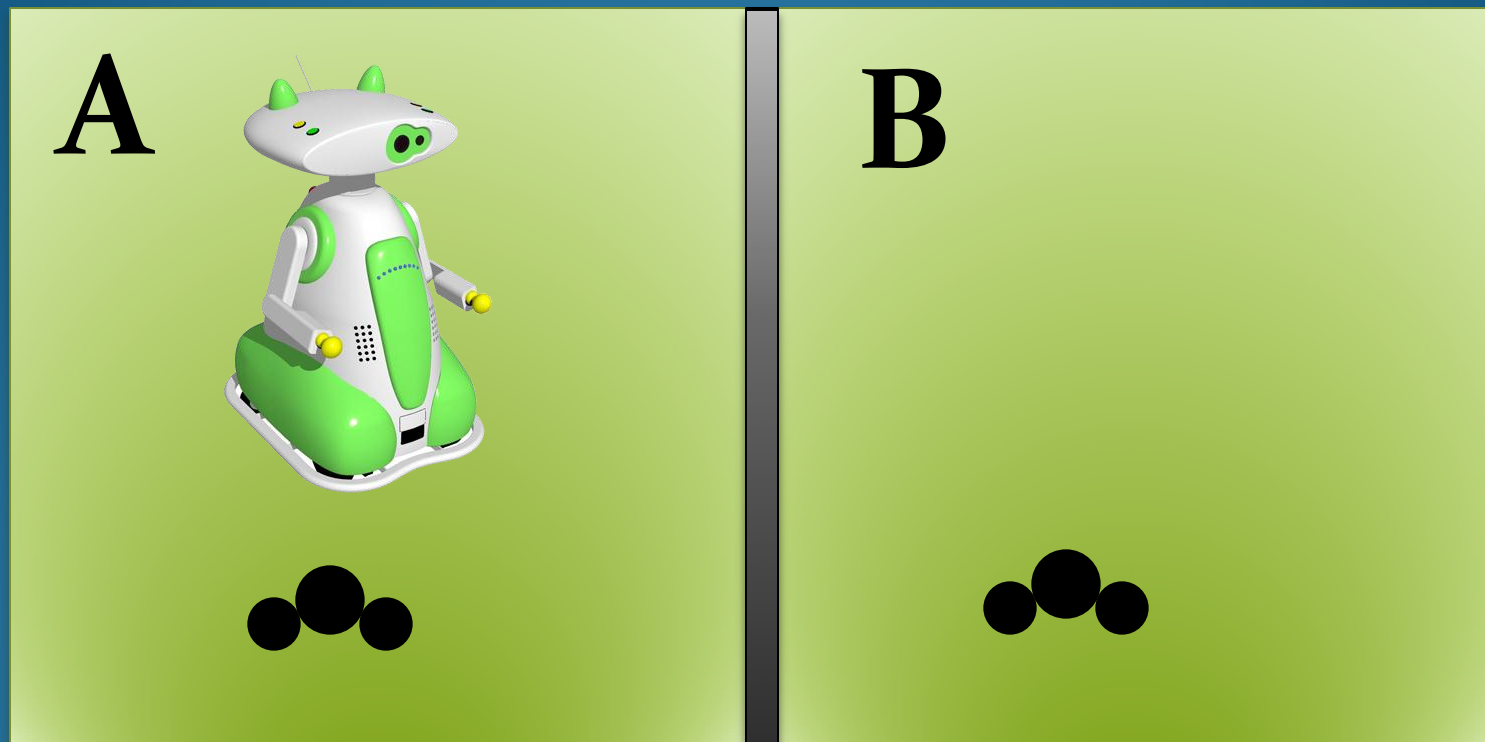


# بحث کنید :

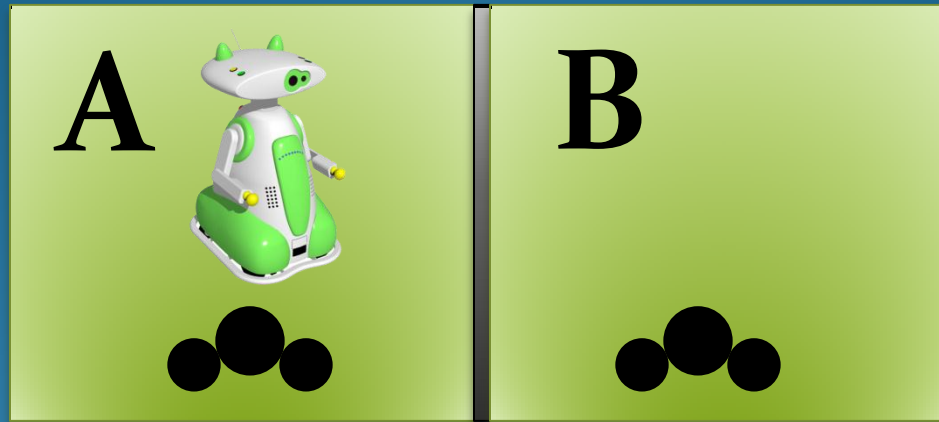
به نظر شما در عامل جارو برقی معیار کارایی را چگونه تعریف کنیم ؟

میزان سر و صدا - میزان مصرف برق - میزان تمیز کردن - زمان تمیز کردن و ...

# بررسی عامل جارو برقی



# بررسی عامل جاربرقی



محیط جارو برقی : تنها شامل دو خانه است که عامل جارو برقی می تواند درک کند در کدام خانه است و هر خانه دو وضعیت دارد یا آشغال دارد یا ندارد .

سه عمل که عامل جارو برقی انجام می دهد :

Left

رفتن به چپ

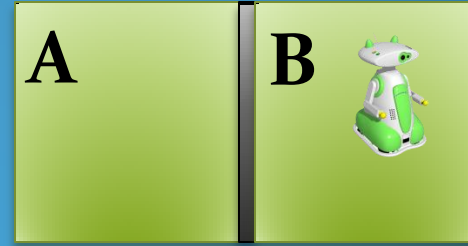
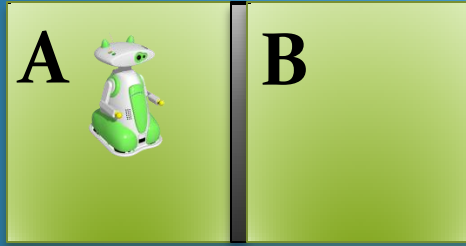
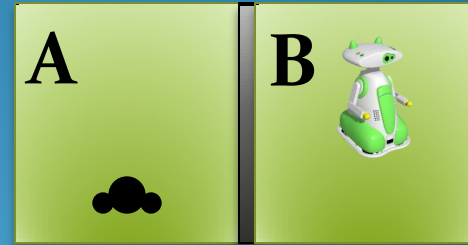
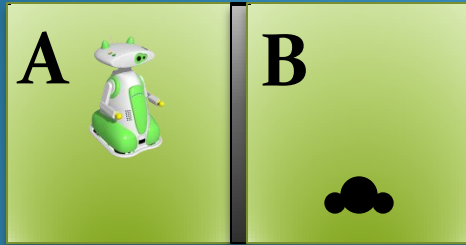
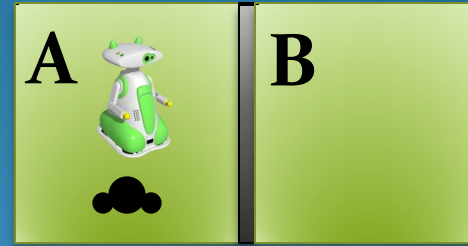
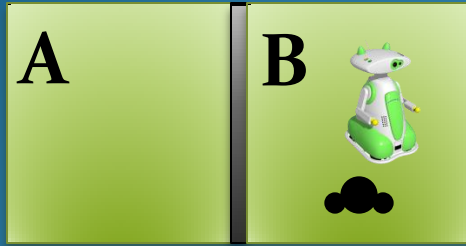
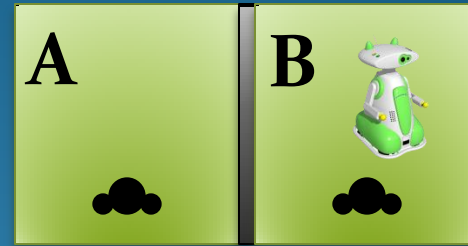
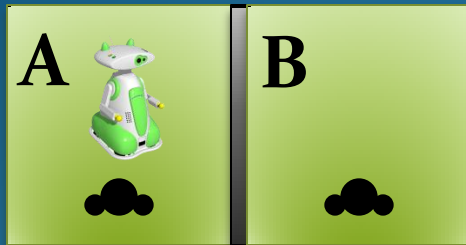
Suck

مکیدن آشغال

Right

رفتن به راست

# بررسی تعداد حالت عامل جارو برقی



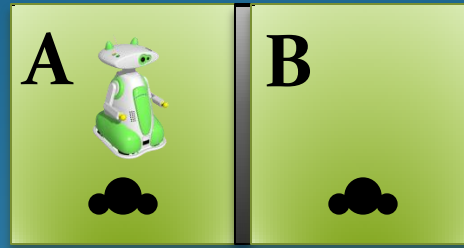


# پرسش؟

به نظر شما در عامل جارو برقی اگر چهار خانه داشته باشیم چند حالت به وجود می آید؟

نشان دهید در حالتی که  $n$  خانه داریم تعداد حالات  $n \times 2^n$

# رشته ادراکات عامل جارو برقی



**رشته ادراک عامل :** هر چیزی که عامل تاکنون درک کرده است (مثل جعبه سیاه هواپیما)  
در واقع انتخاب یک عمل توسط عامل وابسته به رشته ادراکی است که عامل تا آن لحظه  
گردآوری کرده است.

چند نمونه از رشته ادراک عامل جارو برقی :

[A ,Clean]

[A ,Dirty]

...

[A ,Clean] , [B,Dirty]



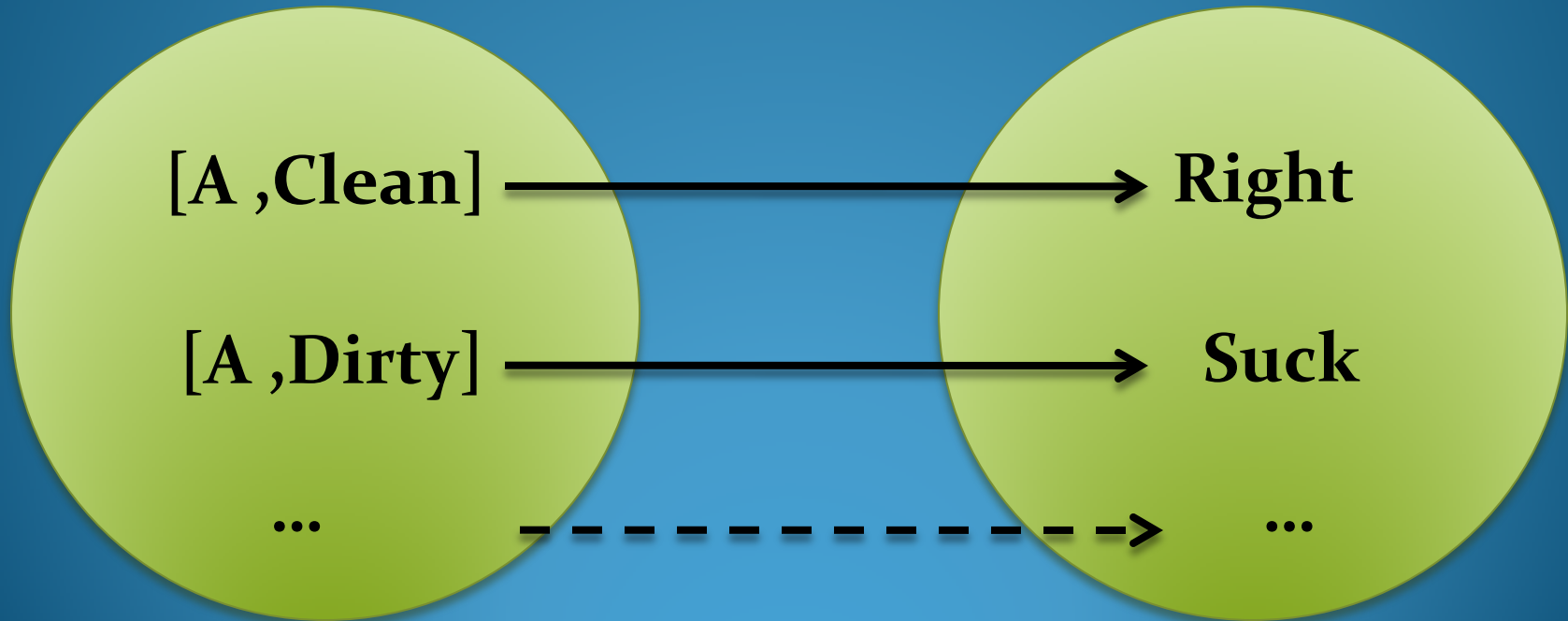
# تابع عامل جاربرقی

**تابع عامل چیست :** اگر بتوانیم بین هر رشته ادراک و عمل مناسب آن رشته ادراک تناظری برقرار سازیم در واقع موفق شده ایم بین مجموعه رشته ادراک و عمل مناسب آن رشته ادراک یک توصیف ریاضی داشته باشیم . این کار توسط تابع عامل انجام میگیرد .

تناظر بین رشته ادراک و مجموعه اعمال جارو برقی

مجموعه رشته ادراک جار برقی

مجموعه اعمال جار برقی





# پرسش؟

به نظر شما در عامل جارو برقی رشته ادراک های زیر چه معنی دارند و عمل مناسب آن را بیابید؟

[A ,Clean] , [A ,Clean] , [A ,Clean]

[A ,Clean] , [A ,Clean] , [A ,Dirty]

# تابع عامل را به چه شکلی ذخیره کنیم

تابع عامل را معمولاً به شکل یک جدول نگهداری می کنند

مجموعه اعمال	مجموعه رشته ادراک
Right	[A ,Clean]
Suck	[A ,Dirty]
Left	[B ,Clean]
Suck	[B ,Dirty]
...	...
Right	[A ,Clean] , [A ,Clean] , [A ,Clean]

قسمتی از جدول یک تابع عامل ساده برای دنیای جاروبرقی

# منطقی بودن عامل

بین منطقی بودن و دانش کامل باید تفاوت قائل شویم .

مثال : عبور از خیابان یا افتادن در چاله !

سه مطلب مهم :

(۱) منطقی بودن عامل باید در حیطه دانشی که دارد سنجیده شود .

(۲) نمی توانیم عاملی را به خاطر شکست در انجام عملی که قادر به انجام آن نیست سرزنش کنیم .

(۳) منطقی بودن به معنای دانش کامل نیست .

# تعریف عامل منطقی ( عقلانی ) ایده آل

یک عامل را **منطقی ایده آل** می گویند که بر اساس دانش درونی خود و رشته ادراکش تا این لحظه عملی را انجام دهد که معیار کارایی خود را ماکزیمم نماید .

عامل منطقی سعی می کند معیار کارایی مورد انتظار از خود را ماکزیمم نماید .

# منطقی بودن عامل به چه چیزهایی بستگی دارد ؟

منطقی بودن عامل ( برای موفقیت بیشتر ) به عوامل زیر بستگی دارد :

## ۱) معیار کارایی

مسلماً یک عامل در راستای معیار کارایی که دارد میتواند منطقی کار کند مثلاً در جارو برقی انتظار نداریم که جارو برقی کاری بجز تمیز کردن انجام دهد ( منطقی نیست ) زیرا معیار کارایی تمیز کردن است .

## ۲) رشته ادراک عامل تا این لحظه

مسلماً انتظار منطقی بودن زمانی است که عامل بر اساس رشته ادراک یک عمل برای انجام داشته باشد .

## ۳) دانش قبلی عامل

اگر طراح عامل در ابتدا مقداری دانش درونی درون عامل قرار دهد .

## ۴) اعمالی که عامل می تواند انجام دهد

عاملی منطقی است که براساس مشاهدات و دانش درونی و یاد گیری خود بتواند کاری انجام دهد.



# یادگیری و خودمختاری یک عامل

به نکات زیر توجه کنید :

رفتار یک عامل هم می تواند بر اساس دانش درونی باشد و هم بر اساس تجربه ناشی از مشاهدات .

عامل منطقی باید خود مختار باشد . برای اینکه دانش ناقص قبلی خود را کامل کند یا حتی تصحیح نماید .

عاملی که تجربه ندارد یا تجربه کمی دارد در ابتدا بطور کاملاً خود مختار عمل می کند و لذا مجبور به یک سری اعمال تصادفی است . این کار به خاطر کسب دانش انجام می گیرد .

عاملی که بعد از مدتی تجربه کافی کسب کرد می تواند تا حد زیادی به دانش درونی خود متکی باشد .

یک عامل خوب = دانش درونی + قدرت یادگیری + کسب تجربه

# یادگیری و خودمختاری یک عامل

یک عامل منطقی باید علاوه بر جمع آوری اطلاعات باید بتواند از مشاهدات خود یاد بگیرد چنین عاملی موفق تر است . مثلاً عامل جارو برقی باید بداند در کدام مکان ها آشغال بیشتری ریخته می شود تا به آنجا بیشتر سرکشی کند ( کشف کردن )

**دانش یا درونی است یا بر اساس مشاهده و یادگیری حاصل می شود .**

اگر تصمیمات عامل تنها بر اساس دانش درونی و اولیه باشد و توجهی به مشاهدات نداشته باشد می گوئیم این عامل **فاقد خود مختاری** است .

**مثال : یک ساعت غیر هوشمند**

اگر تصمیمات عامل علاوه بر دانش درونی متکی به مشاهدات نیز باشد می گوئیم این عامل **خود مختاری** است .

**مثال : یک ساعت هوشمند**

# انواع محیط اطراف عامل

(۱) قابل دسترسی و غیر قابل دسترسی

**محیط قابل دسترسی :** محیطی که عامل آن توسط ابزار حس کننده‌اش امکان دسترسی به وضعیت کامل محیط را داشته باشد.  
محیط قابل دسترسی راحت است، زیرا عامل نیازمند دستکاری هیچ وضعیت داخلی برای حفظ دنیا را نخواهد داشت.

# انواع محیط اطراف عامل

## (۲) قطعی و غیر قطعی

**محیط قطعی :** محیطی است که اگر وضعیت بعدی محیط بوسیله وضعیت کنونی و اعمالی که با عامل‌ها انتخاب گردد، تعیین شود. بهتر است به قطعی یا غیر قطعی بودن محیط از دیدگاه عامل نگاه کنیم.

**دنیای جارو برقی قطعی و دنیای راننده تاکسی غیر قطعی است .**

# انواع محیط اطراف عامل

(۳) ایستا و پویا

**محیط پویا:** محیطی که در حین سنجیدن عامل تغییر می کند. مثل راننده تاکسی

**محیط نیمه پویا:** محیطی که با گذر زمان تغییر نمی کند اما امتیاز کارایی تغییر می کند. مثل شطرنج

**محیط ایستا:** محیطی که در حین سنجیدن عامل تغییر نمی کند.

مثل جدول کلمات متقاطع

محیط های ایستا برای کار ساده هستند زیرا عامل نیاز به نگاه کردن به

دنیا در حین تصمیم گیری عملی نداشته و همچنین در مورد گذر

زمان نیز نگران نمی باشد.

# انواع محیط اطراف عامل

۴) گسسته و پیوسته

**محیط گسسته** : اگر تعداد محدود و مجزا از ادراک و اعمال بوضوح تعریف شده باشد.

مثال محیط گسسته : بازی شطرنج

مثال محیط پیوسته : راننده تاکسی

**سخت‌ترین حالت در بین حالات موجود برای محیط:**

غیر قابل دسترسی و پویا و پیوسته



# بررسی عامل راننده تاکسی



محیط	اهداف	عملیات	ادراکات	نوع عامل
جاده، پیاده‌رو، ترافیک، مشتری	ایمنی، سرعت، قانونمندی، راحتی، افزایش سودمندی	راهنمایی کردن، شتاب‌دهنده، ترمز، صحبت با مسافر	دوربین‌ها، سرعت سنج، GPS، Sonar میکروفون	راننده تاکسی

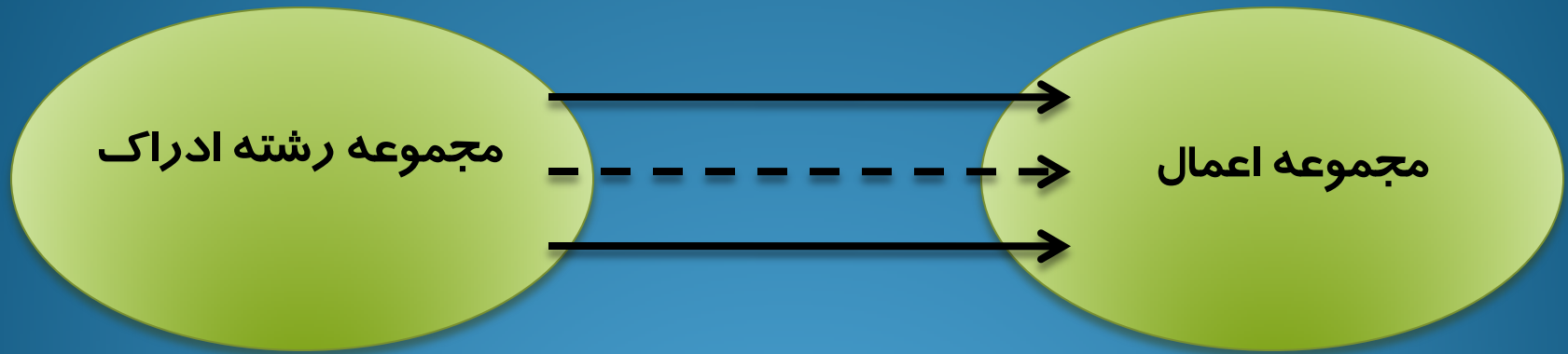
# بررسی محیط چند عامل مختلف

محیط	قابل دسترسی	قطعی	ایستا	گسسته
شطرنج به همراه ساعت	YES	YES	نیمه پویا	YES
شطرنج بدون ساعت	YES	YES	YES	YES
پوکر	NO	NO	YES	YES
تخته نرد	YES	NO	YES	YES
راندن تاکسی	NO	NO	NO	NO
سیستم تشخیص پزشکی	NO	NO	NO	NO
ربات جابجا کننده اشیاء	NO	NO	NO	NO
کنترل کننده پالایشگاه	NO	NO	NO	NO
آموزش دهنده انگلیسی با ارتباط متقابل	NO	NO	NO	YES

# بررسی ساختار عامل های هوشمند

## توجه :

تابع عامل یک رشته مشاهدات را به یک سری اعمال نگاشت می کند .



برنامه عامل به برنامه ای می گویند که تابع عامل را پیاده سازی کند .

نتیجه گیری : بین برنامه عامل و تابع عامل باید تفاوت قائل شویم

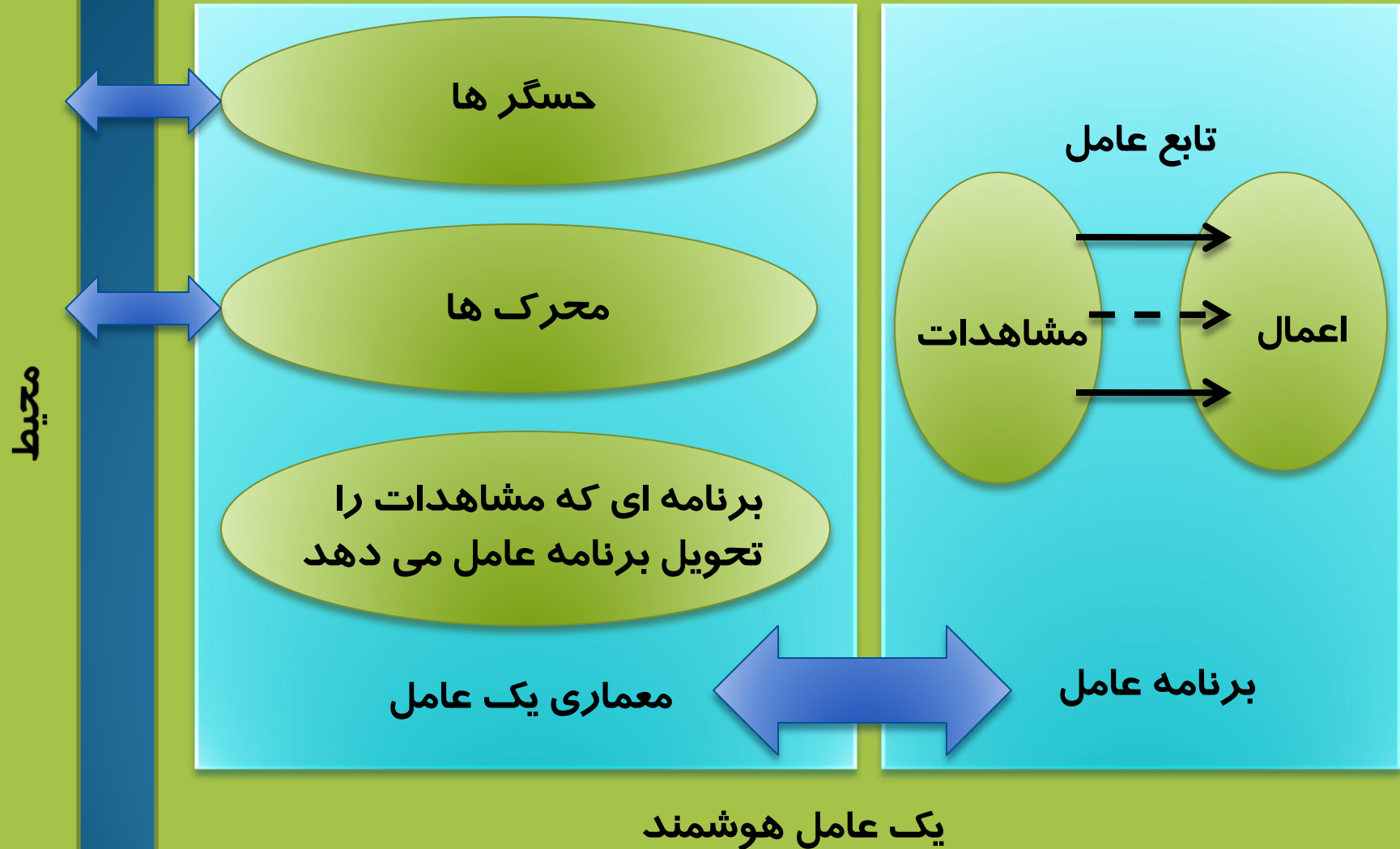
# معماری عامل های هوشمند

## معماری یک عامل چیست ؟

معماری عامل مشاهداتی را که از طریق حسگر دریافت می کند تحویل برنامه عامل می دهد .

پس معماری عامل می تواند سخت افزاری باشد مثل یک فیلتر کننده صدا یا یک چشم الکترونیک باشد یا حتی می تواند به شکل یه نرم افزار باشد که روی یک سخت افزار خاص پیاده شده باشد مثل یک نرم افزاری که روی یک دروین قرار دارد و تصاویر دریافتی را تحویل برنامه عامل می دهد .

# رابطه بین معماری و برنامه عامل



# رابطه بین معماری و برنامه عامل

معماری عامل + برنامه عامل = عامل

هدف طراح هوش مصنوعی طراحی **برنامه عامل** است .





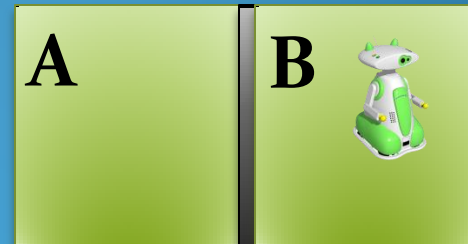
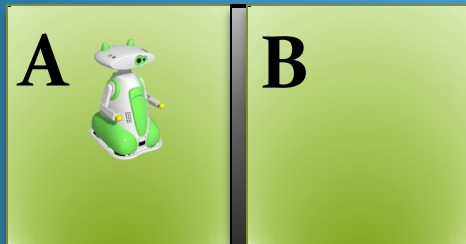
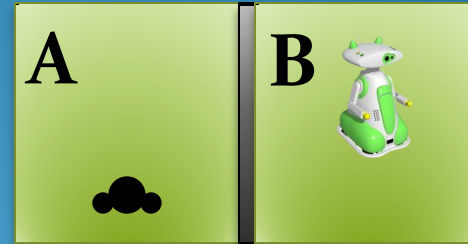
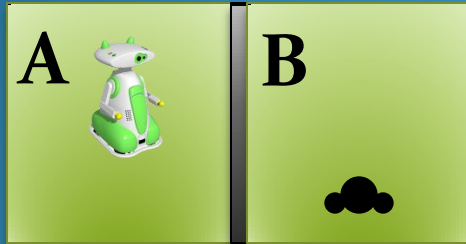
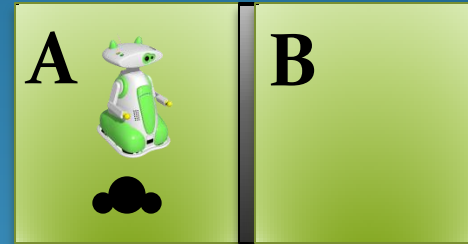
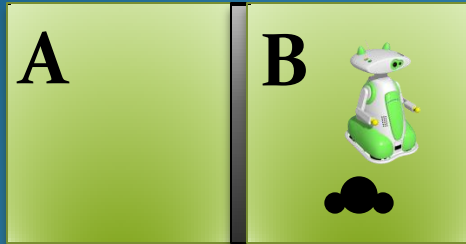
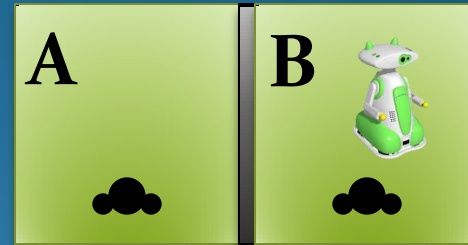
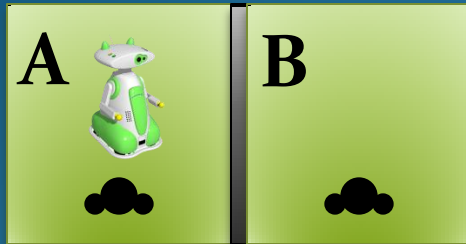
# بحث

در مورد برنامه عامل و تابع عامل و معماری  
عامل هوشمند جارو برقی بحث کنید ؟

# بررسی برنامه عامل جارو برقی

```
Public Action MyAgent( Location , State)
{
    if ( State==Dirty )
        return Suck ;
    else if ( Location == Left)
        return Right;
    else if ( Location == Right)
        return Left;
}
```

# بررسی مجدد عامل جارو برقی



# تابع عامل جارو برقی در محیط دو خانه ای

Location	State	Action
A	Dirty	Suck
A	Clean	Right
B	Dirty	Suck
B	Clean	Left

← تابع عامل

Public Action MyAgent( Location , State)

{

if ( State==Dirty )

return Suck ;

else if ( Location == Left)

return Right;

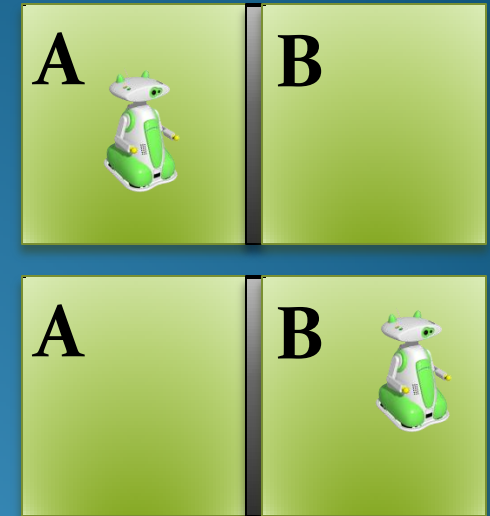
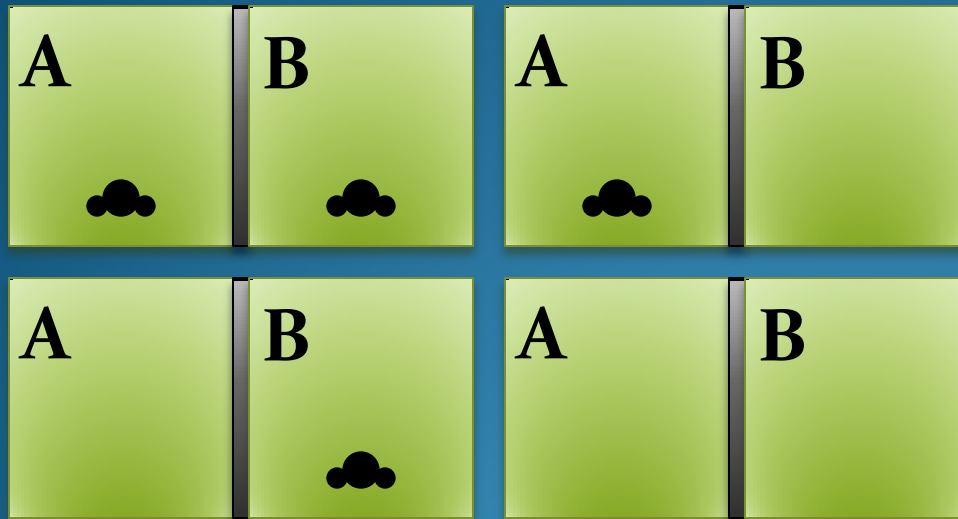
else if ( Location == Right)

return Left;

}

← برنامه عامل

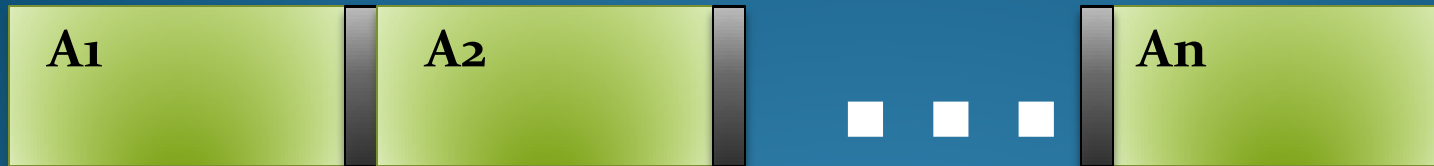
# تعداد حالات جارو برقی در محیط دو خانه



تعداد حالات قرارگرفتن جارو برقی × تعداد حالات ریختن زباله بودن خانه ها

$$2 \times 2 \times 2 = 2^2 \times 2$$

# تعداد حالات جارو برقی در محیط n خانه



زباله دارد یا ندارد

زباله دارد یا ندارد

زباله دارد یا ندارد

$$2 \times 2 \times \dots \times 2 = 2^n$$

تعداد حالات قرارگرفتن جارو برقی  $n =$

$$n \times 2^n = \text{تعداد حالات کلی}$$

# تعداد حالات جارو برقی در محیط سه خانه



۲۴ حالت

$$n = 3$$

$$\text{تعداد حالات} = n \times 2^n = 3 \times 2^3 = 24$$

# راهنمایی برای پروژه جاروبرقی

Location	State	Action
A	Dirty	Suck
A	Clean	Go To B
B	Dirty	Suck
B	Clean	Go To C
B	Clean	Go To A
C	Dirty	Suck
C	Clean	Go To B

تابع عامل ←

Public Action MyAgent( Location , State)

```
{  
    if ( State==Dirty )  
        return Suck ;  
    else if ( Location == A)  
        return B ;  
    else if ( Location == B)  
        return A or C ;  
    else if ( Location == C)  
        return B ;  
}
```

برنامه عامل ←



# شکل تابع عامل جاروبرقی

Location	State	Action
A	Dirty	Suck
A	Clean	Go To B
B	Dirty	Suck
B	Clean	Go To C
B	Clean	Go To A
C	Dirty	Suck
C	Clean	Go To B

← تابع عامل به شکل انتزاعی

0	1	0
0	0	1
1	1	0
1	0	2
1	0	3
2	1	0
2	0	1

یک نمونه ساختمان داده برای تابع عامل



# شکل برنامه عامل برای جاروبرقی

0	1	0
0	0	1
1	1	0
1	0	2
1	0	3
2	1	0
2	0	1

Location	State	Action
A	Dirty	Suck
A	Clean	Go To B
B	Dirty	Suck
B	Clean	Go To C
B	Clean	Go To A
C	Dirty	Suck
C	Clean	Go To B

تابع عامل به شکل ماتریس

تابع عامل به شکل انتزاعی

```
if ( matrix[ i ][ 1] == 1)
{
    action= 0 ;
}
else if ( matrix[ i ][ 1] == 0 && matrix[ i][0] == 0)
{
    action= 1 ;
}
... و
```

قسمتی از کد برنامه عامل به  
زبان جاوا



# سوال

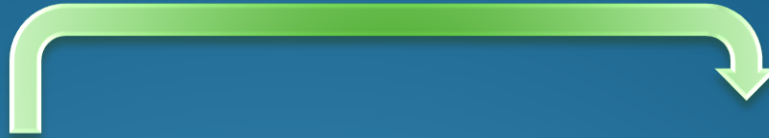
چگونه می توانیم حجم تابع عامل ( سایز جدول ) را کاهش دهیم ؟

Location	State	Action
A	Dirty	Suck
A	Clean	Go To B
B	Dirty	Suck
B	Clean	Go To C
B	Clean	Go To A
C	Dirty	Suck
C	Clean	Go To B

# پاسخ :



## بعد از کاهش حجم پیدا کردن



Location	State	Action
A	Dirty	Suck
A	Clean	Go To B
B	Dirty	Suck
B	Clean	Go To C
B	Clean	Go To A
C	Dirty	Suck
C	Clean	Go To B

Location	State	Action
A or B or C	Dirty	Suck
A	Clean	Go To B
B	Clean	Go To C
B	Clean	Go To A
C	Clean	Go To B

یعنی : وقتی که خانه ای کثیف است بررسی نکن که کجا هستی

# سوال

به چه علتی عامل هایی که جدول گرا هستند محکوم به شکست هستند؟



# پاسخ :

- ۱) اکثر واقع سائز جدول بسیار بزرگ است ( تعداد سطر زیادی دارد )
- ۲) بعضی از جدول ها آنقدر بزرگ هستند که نگهداری آنها در حافظه ممکن نیست
- ۳) ساختن جدول وقت زیادی ازطراح هوش مصنوعی میگیرد .
- ۴) جدول در واقع نقش جعبه سیاه و تجربیات عامل را دارد . و هیچ عاملی نمی تواند تنها بر اساس تجربیات یاد بگیرد .
- ۵) نحوه پر کردن اطلاعات در جدول ساده نیست .

# مثال از استفاده از راهبرد جدول گرا در ریاضیات

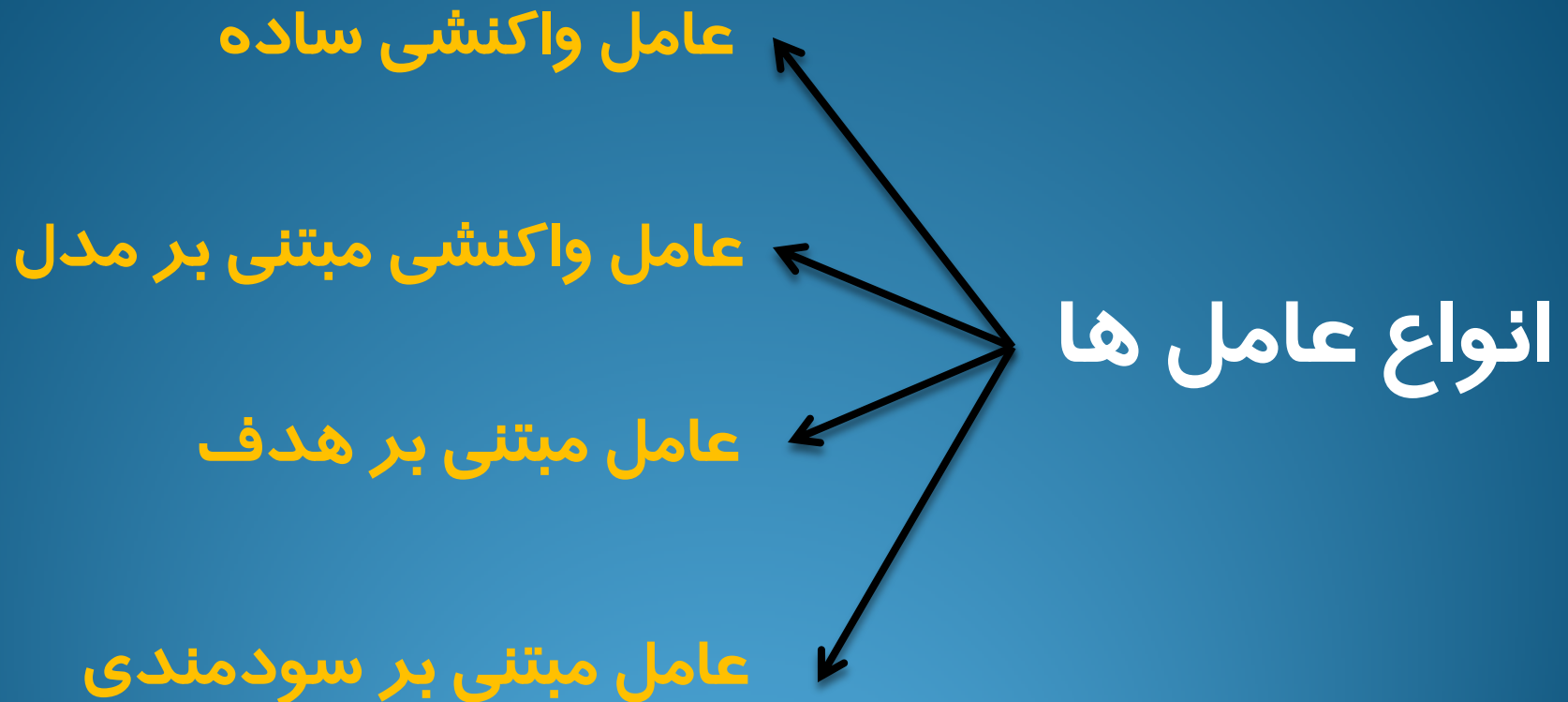
## محاسبه جذر اعداد بوسیله جدول :

عدد	جذر
۲	۱.۴
۳	۱.۷
۵	۲.۲۳
۶	۲.۴۴
...	...

راهبرد هوشمند برای محاسبه جذر یک عدد ؟  
روش نیوتن - رافسون یا نصف کردن معادله زیر :

$$x^2 - C = 0$$

# انواع عامل ها





# عامل واکنشی ساده

ساده ترین نوع عامل = عامل واکنشی ساده

اعمال بر اساس مشاهده فعلی صورت می گیرد

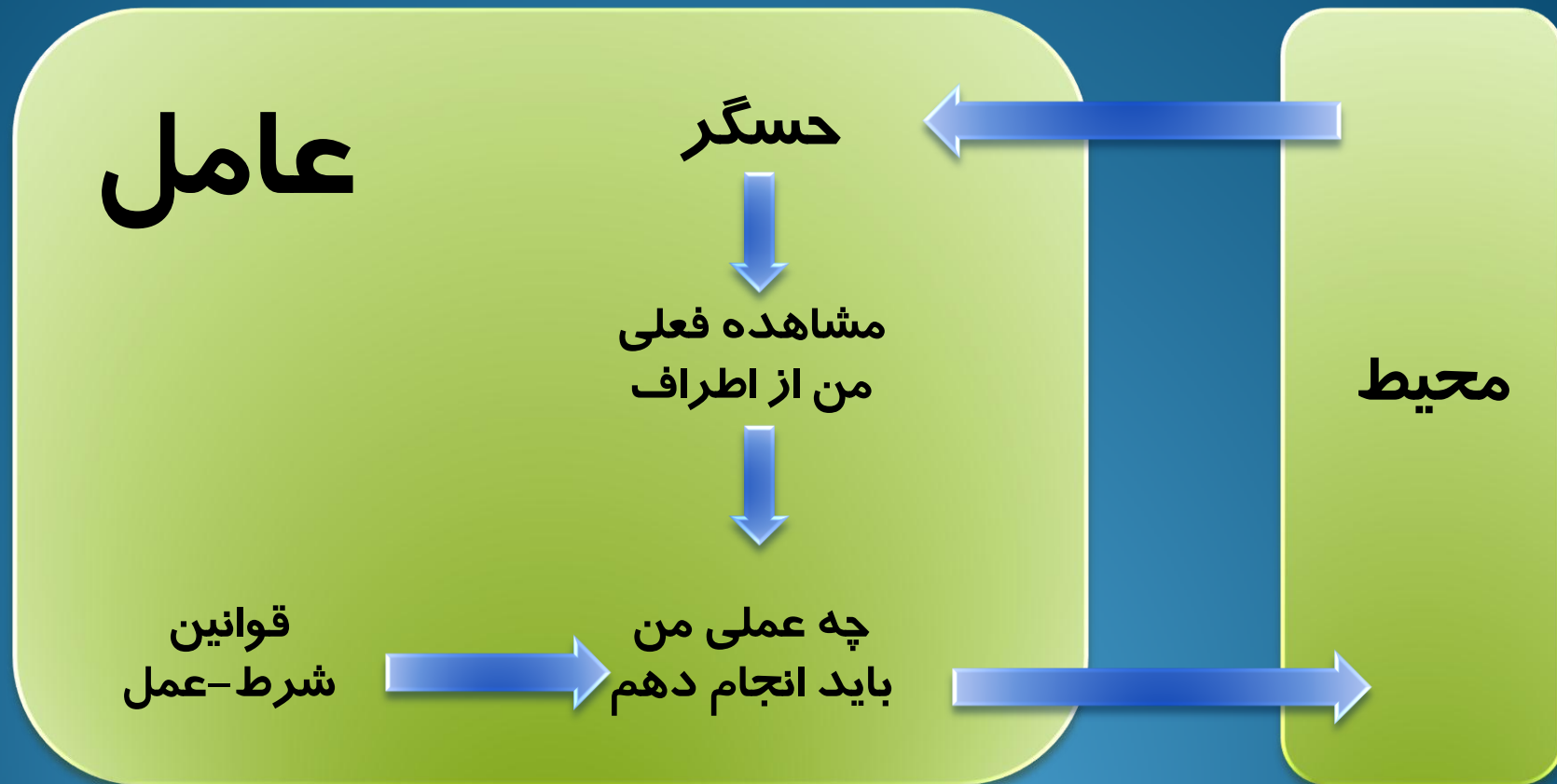
تابع عامل آنها ساده است و به شکل یک جدول است .

عامل واکنشی ساده از نظر سادگی قابل ستایش و از نظر هوشمندی ضعیف هستند .

به خاطر حذف سابقه ادراک برنامه عامل در مقایسه با جدول آن بسیار کوچک است

مثال : عامل جارو برقی

# ساختار عامل واکنشی ساده



عامل واکنشی ساده



# سوال

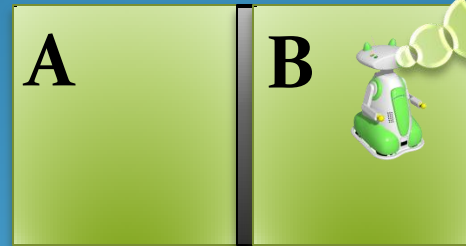
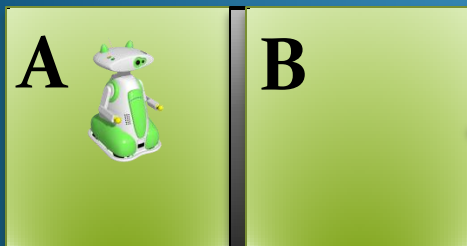
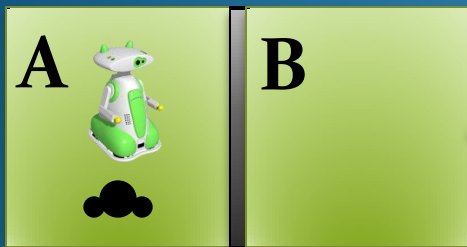
اگر عامل جارو برقی دارای یک چشم الکترونیک باشد که  
نرخ ورودی آن ۵۰ مگا بایت بر ثانیه باشد نشان دهید که  
در یک ساعت مشاهده به مقدار زیر جدول عامل ردیف  
دارد :

$260 \times 60 \times 50M$



# سوال

با توجه به شکل زیر چرا عامل جاروبرقی واکنشی ساده است ؟



من قبلاً در کدام حالت بوده ام  
؟! تنها می دانم آلان در کجا و در  
چه وضعیتی هستم

# سوال

آیا عاملی که به شکل lookup Table پیاده سازی می شود می تواند خودمختار باشد ؟ چرا ؟

# توجه !



عامل های واکنشی ساده به چند علت خوب عمل نمی کنند :

( ۱ ) ایجاد جدول نگاشت حالت - عمل همیشه ممکن نیست  
و گاهی اوقات زمانبر است .

( ۲ ) گاهی اوقات جدول نگاشت حالت - عمل تعداد سطر  
زیادی دارد مثلاً بازی شطرنج تقریباً <sup>100</sup>35 سطر دارد .

( ۳ ) این عامل اگر حتی قدرت یادگیری داشته باشد  
زمان زیادی صرف به روز رسانی جدول می شود .

( ۴ ) این عامل فاقد هرگونه خود مختاری است .

( ۵ ) فقط حال را می بیند نه آینده نه گذشته

علت

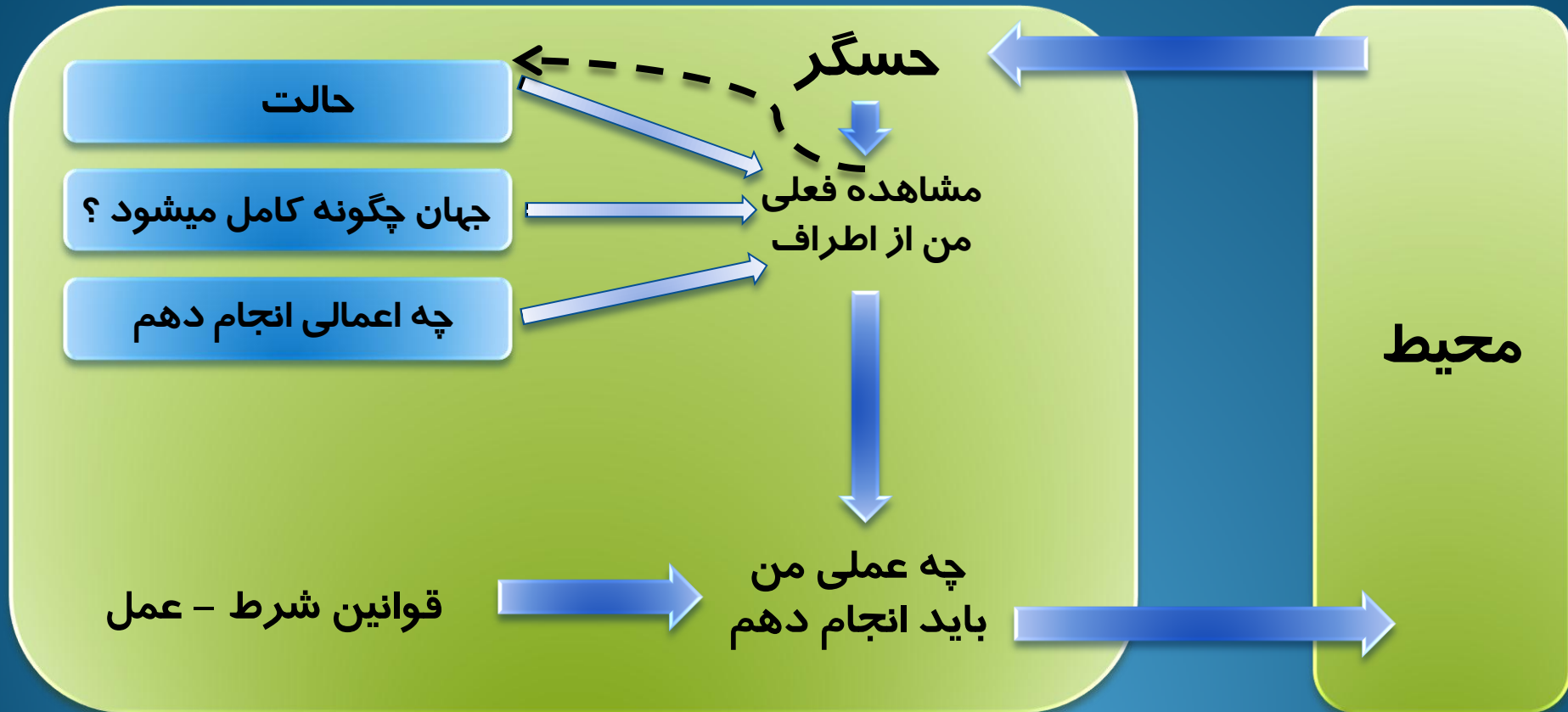
( ۶ ) در شرایط غیر منتظره کارایی ندارد (فقدان خود مختاری)

# عامل واکنشی مبتنی بر مدل

این عامل ها تغییرات محیط را دنبال میکنند .

مثال : یک راننده که برای سبقت گرفتن بایستی لحظه به لحظه تغییرات محیط را بررسی کند.

# ساختار عامل مبتنی بر مدل



عامل مبتنی بر مدل



# عامل واکنشی مبتنی بر هدف

این عامل وضعیت هایی را به عنوان هدف در نظر میگیرند.  
و سپس به دنبال رشته ای از Action ها می گردند تا به هدف  
برسد .

عامل ها برای تصمیم گیری درست علاوه بر دانستن وضعیت  
کنونی محیط به نوعی هدف نیاز دارند .

عامل های هدفمند گاهی اوقات نیازمند بررسی دنباله ای از  
رشته Action ها می گردند تا به هدف برسند . این فرایند در  
هوش شاخه ای را بنام جستجو ایجاد کرده است .

# عامل واکنشی مبتنی بر هدف

عامل های هدف گرا آینده نگر هستند .

عامل های هدف گرا ظاهراً کم کارآمد ولی بسیار انعطاف پذیرند.

عامل های هدف گرا با تغییر هدف رفتار خود را عوض می کنند.



یک عامل هدف گرا که هدف آن رسیدن به مقصد است

# مقایسه عامل واکنشی و عامل مبتنی بر هدف

عامل های هدف گرا با تغییر هدف رفتار خود را عوض می کنند.



یک عامل هدف گرا

عامل های واکنشی با تغییر هدف برای اینکه رفتار خود را عوض کنند باید جدول آنها یا قوانین شرط - عمل آنها تغییر کند .



یک عامل واکنشی



# توجه !

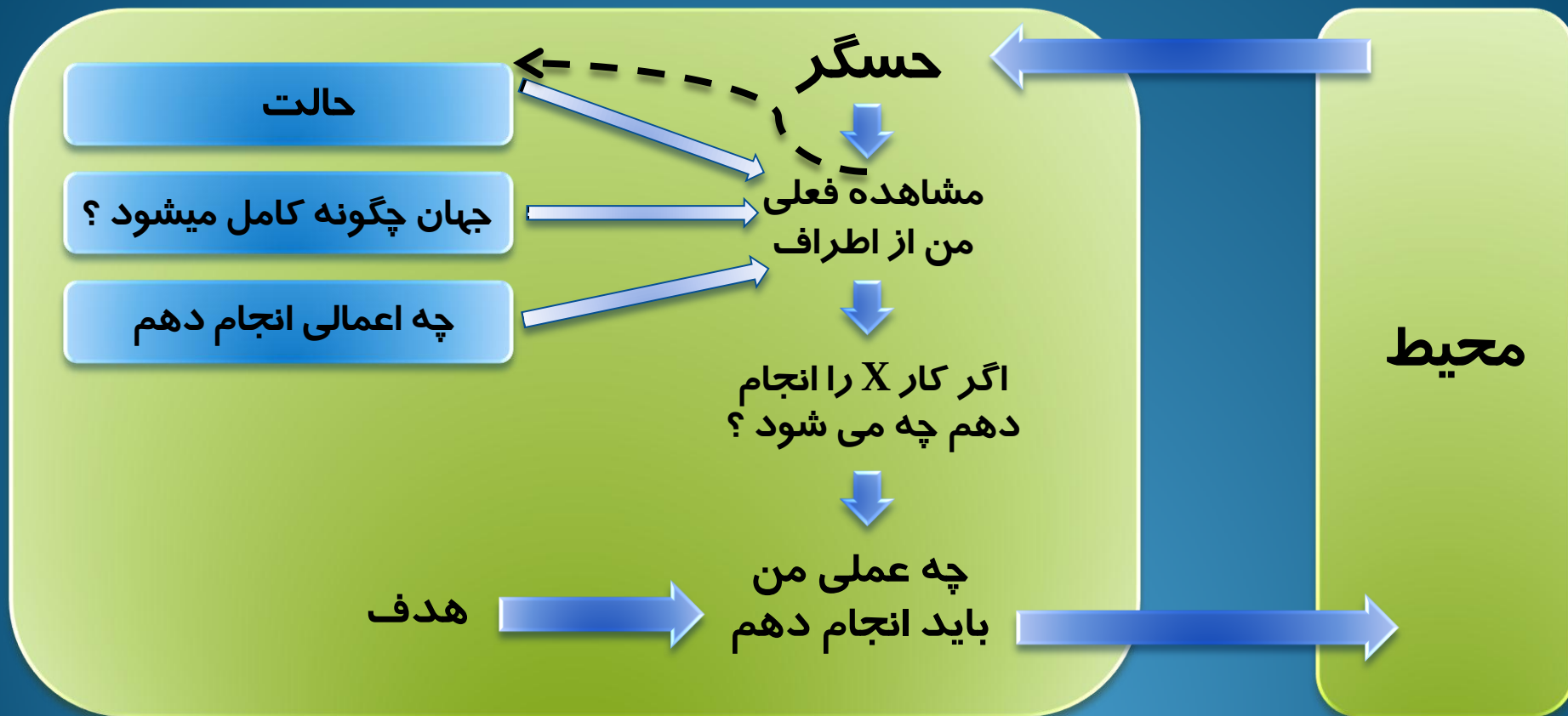
## در یک عامل هدف گرا



## در یک عامل هدف گرا :

گاهی اوقات انتخاب عمل آسان و گاهی اوقات مشکل است  
( به علت چندین عمل کاندید و انتخاب یک رشته اعمال )

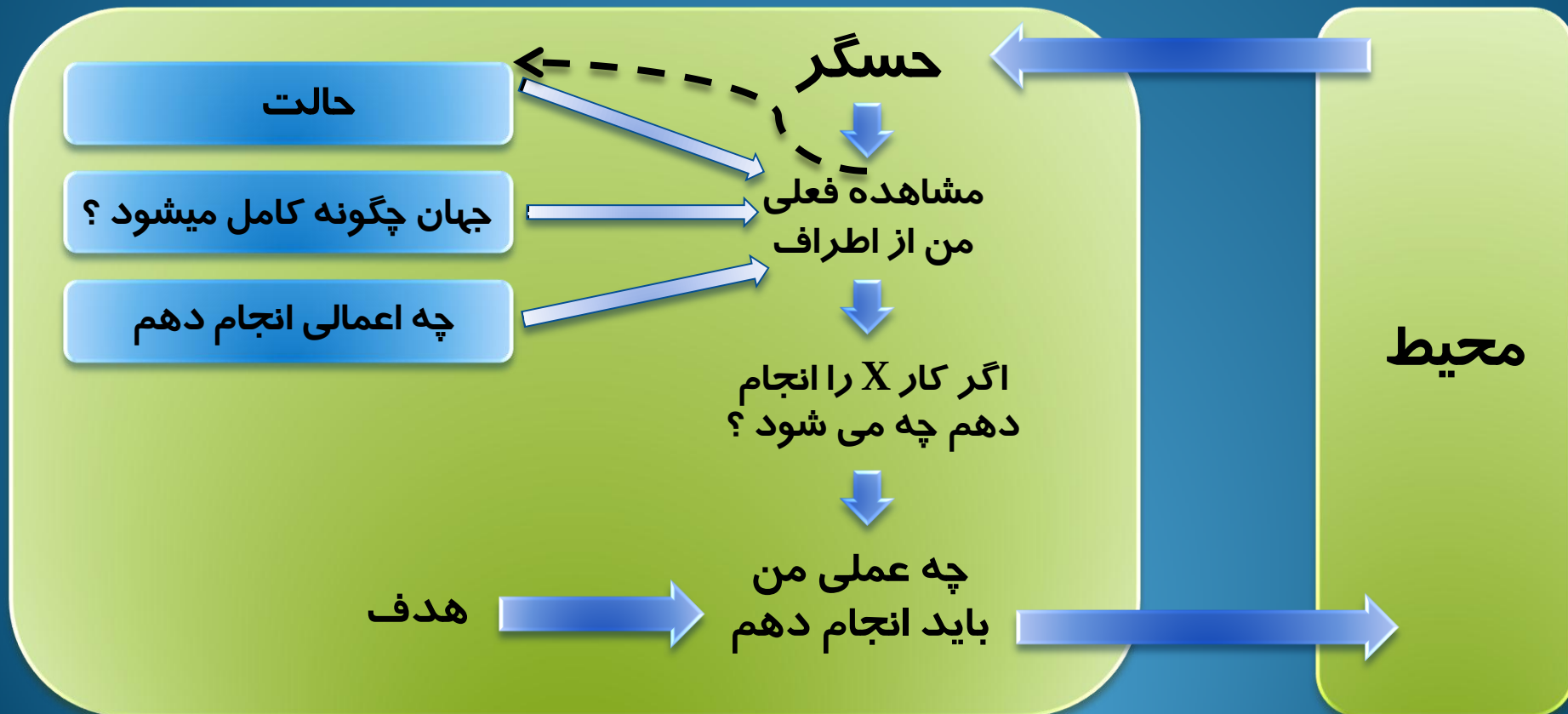
# ساختار عامل هدف گرا



عامل هدف گرا

# سوال

چرا در شکل زیر از فلش چین استفاده شده ؟



# عامل مبتنی بر سودمندی

در این عامل ها وضعیت هایی انتخاب می شود که به هدف نزدیکتر و میزان بهره وری آنان بیشتر است .

در اینجا می خواهیم رفتار با کیفیتی از خود نشان دهیم و به هدف برسیم .

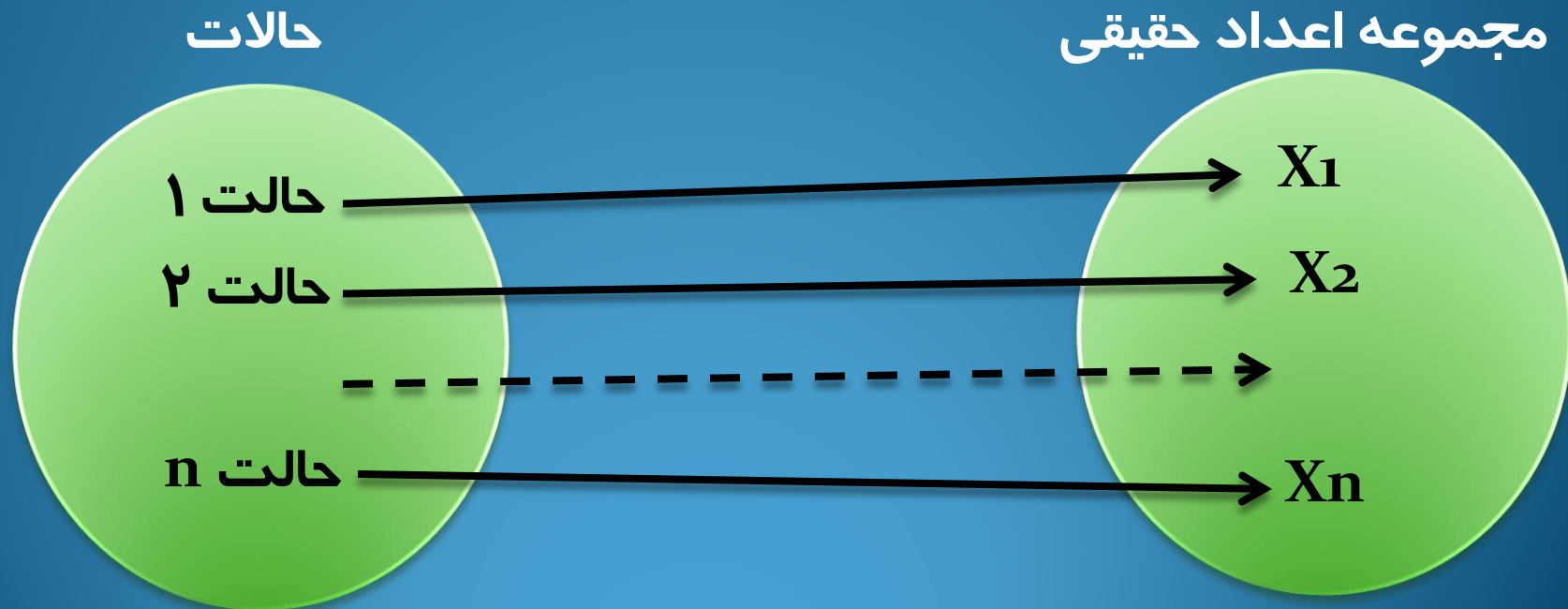
در اینجا دنبال رشته اعمالی هستیم که به هدف برسیم ولی با بهره وری بالا

عامل های سودمند راهی را برای رسیدن به هدف انتخاب می کنند که بیشترین سودمندی را داشته باشد .

# تابع سودمندی چیست ؟

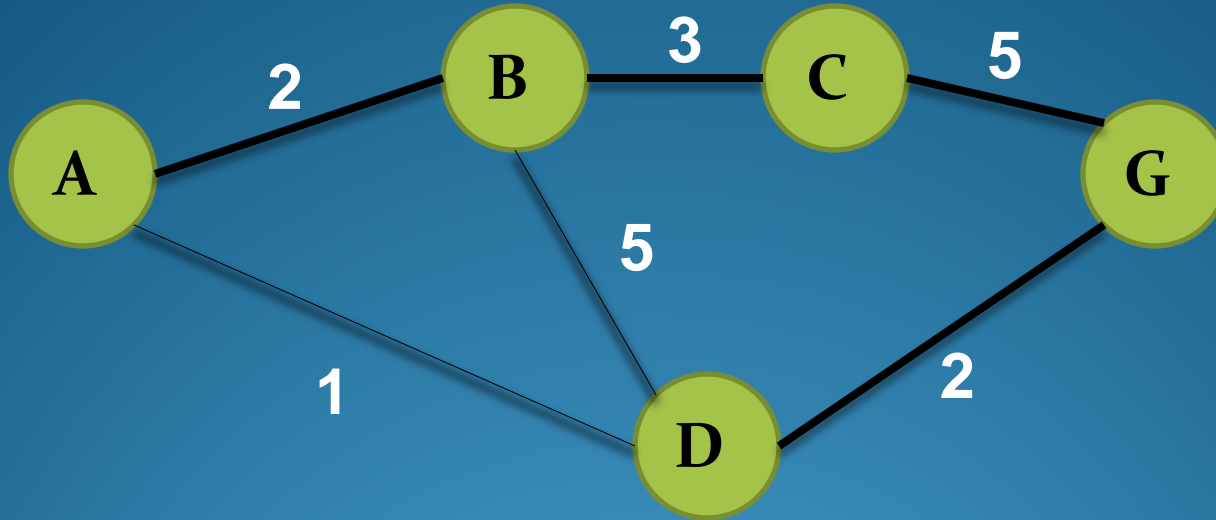
اگر بخواهیم حالتی را بر حالت دیگر ترجیح دهیم از تابع سودمندی استفاده می کنیم .

تابع سودمندی یک حالت را به یک عدد حقیقی نگاشت می کند این عدد در واقع درجه رضایت عامل از آن حالت است .





# یک مثال از تابع سودمندی



حالات

مجموعه اعداد حقیقی

تابع سودمندی



# شکل ریاضی تابع سودمندی

حالات

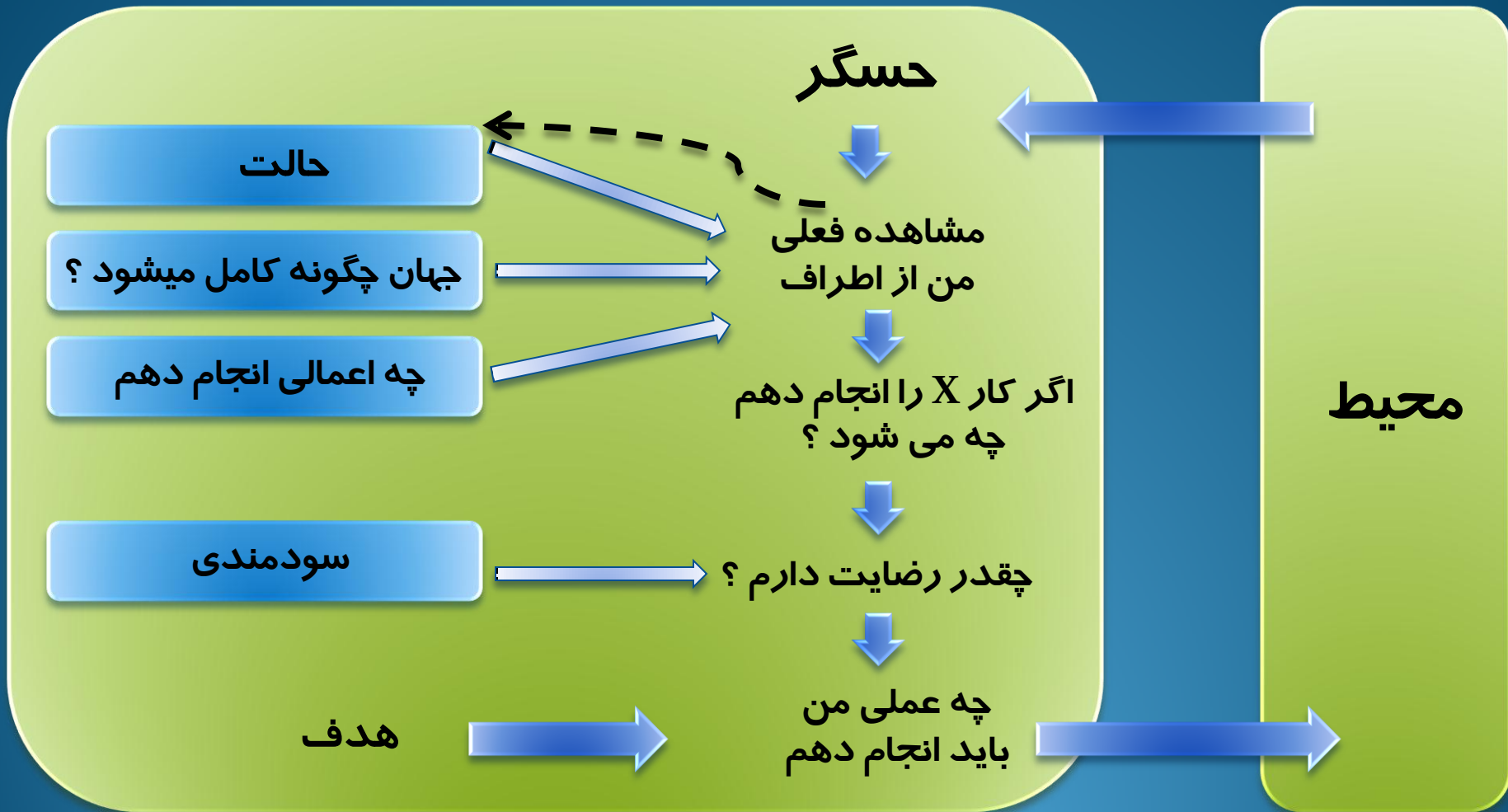
مجموعه اعداد حقیقی

تابع سودمندی



$$\left\{ \begin{array}{l} F(A-B-C-G) = 10 \\ F(A-B-D-G) = 9 \\ F(A-D-G) = 3 \end{array} \right.$$

# ساختار عامل سودمند



عامل سودمند

# عوامل های یادگیرنده

ارائه توسط یک دانشجو  
به شکل تصادفی

مدت ارائه : ۱۰ دقیقه

# موضوعات ارائه هوش مصنوعی

ترجمه تمرینات فصل ۱ کتاب ۱ نفر

ترجمه تمرینات فصل ۲ کتاب گروه ۲ نفری

ترجمه تمرینات فصل ۳ کتاب گروه ۴ نفری

ترجمه تمرینات فصل ۴ کتاب گروه ۴ نفری

ترجمه تمرینات فصل ۵ کتاب گروه ۴ نفری

ترجمه تمرینات فصل ۶ کتاب گروه ۴ نفری

این افراد تمرین ها را به شکل پاور پوینت ارائه می دهند

# موضوعات ارائه هوش مصنوعی

گروه ۲ نفری

گروه ۲ نفری

گروه ۲ نفری

گروه ۲ نفری

گروه ۲ نفری

گروه ۲ نفری

گروه ۲ نفری

گروه ۲ نفری

منطق فازی ۱

منطق فازی ۲

سیستم های خبره

الگوریتم مورچگان

الگوریتم پرندگان

الگوریتم زنبورها

شبکه عصبی

الگوریتم سرد و گرم

# موضوعات ارائه هوش مصنوعی

ژنتیک ۱ نفر

معرفی الگوریتم ژنتیک گروه ۲ نفری

روش های کدینگ در الگوریتم ژنتیک گروه ۲ نفری

روش های انتخاب در الگوریتم ژنتیک گروه ۲ نفری

روش های ترکیب در الگوریتم ژنتیک گروه ۲ نفری

جهش در الگوریتم ژنتیک ۱ نفر

فرموله سازی مسئله در الگوریتم ژنتیک گروه ۲ نفری

ارائه مطالب در نرم افزار پاور پوینت ۲۰۱۰

وقت هر شخص ۱۵ دقیقه

انتخاب موضوع جلسه شنبه مورخ ۱۳۸۹/۲/۴



# پروژه برنامه نویسی ۱



برنامه ای هوشمند بنویسید که یک عامل هوشمند  
جارو برقی با محیط ۳ سلول را شبیه سازی کند . در این  
پروژه ابتدا تابع عامل را به شکل یک جدول پیاده سازی  
کنید و سپس یک برنامه عامل بنویسید که از تابع عامل  
استفاده کند . برنامه شما باید امکاناتی مثل حسگر و  
محرک و ... را شبیه سازی کند .

# یک نمونه برنامه هوشمند جارو برقی به زبان C# توسط اینجانب



# نمره پروژه جارو برقی

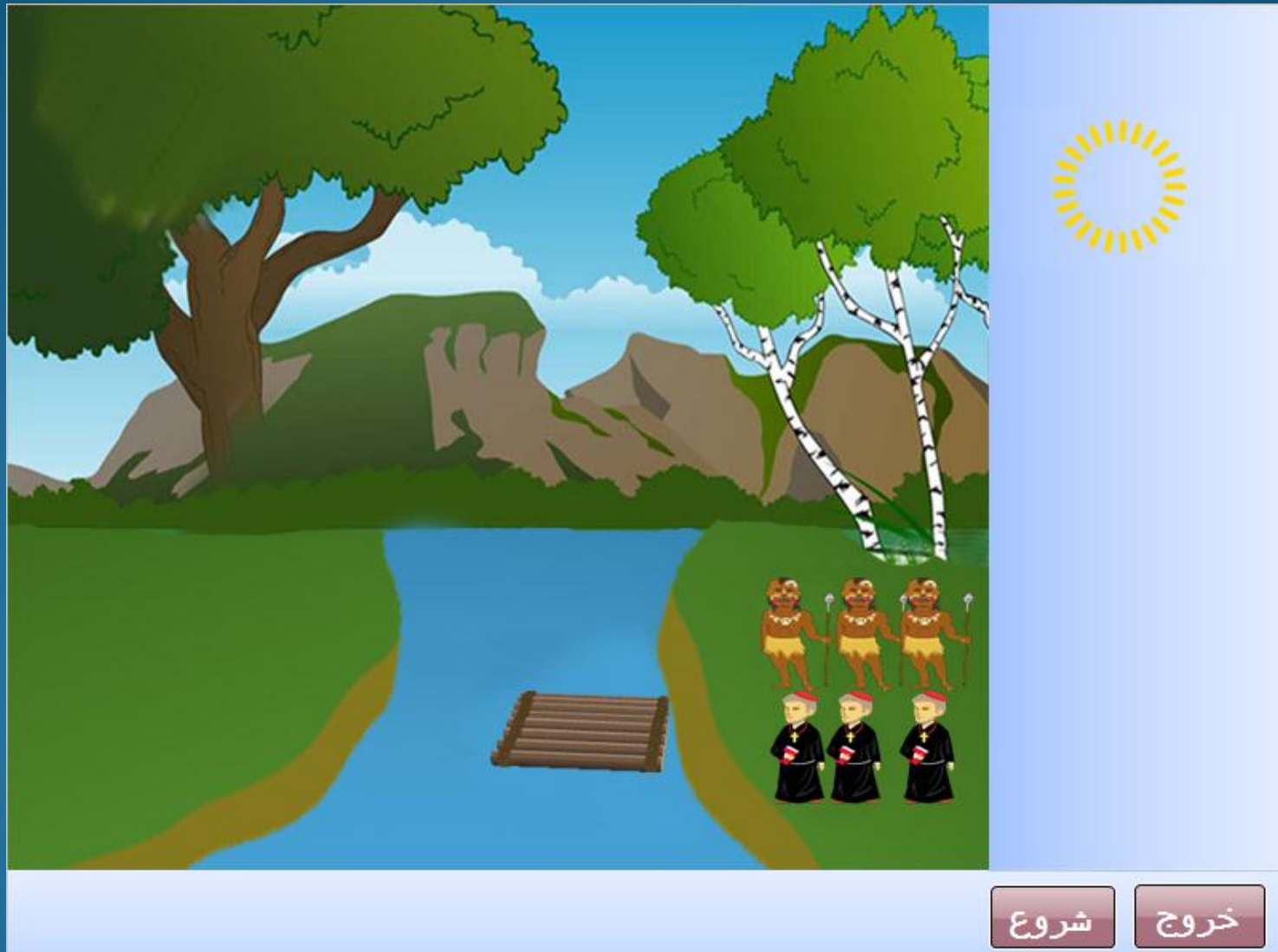
زبان	VC++	۴۵ نمره
زبان	Java	۴۰ نمره
زبان	C#	۳۵ نمره
زبان	C/C++	۳۰ نمره
سایر زبان ها		۲۵ نمره

# پروژه برنامه نویسی ۲

## سه کشتیش و سه آدمخوار



# یک نمونه برنامه هوشمند کشیش و آدمخوار به زبان C# توسط اینجانب

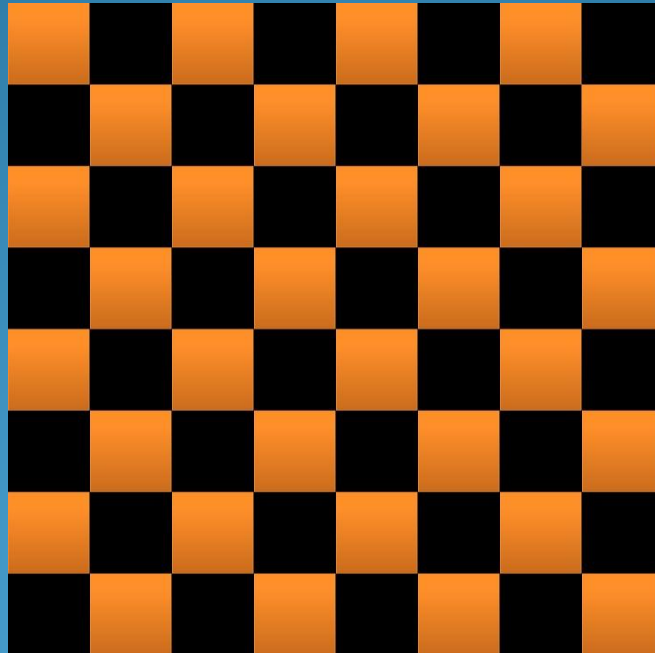


# نمره پروژه سه کشیش و سه آدمخوار

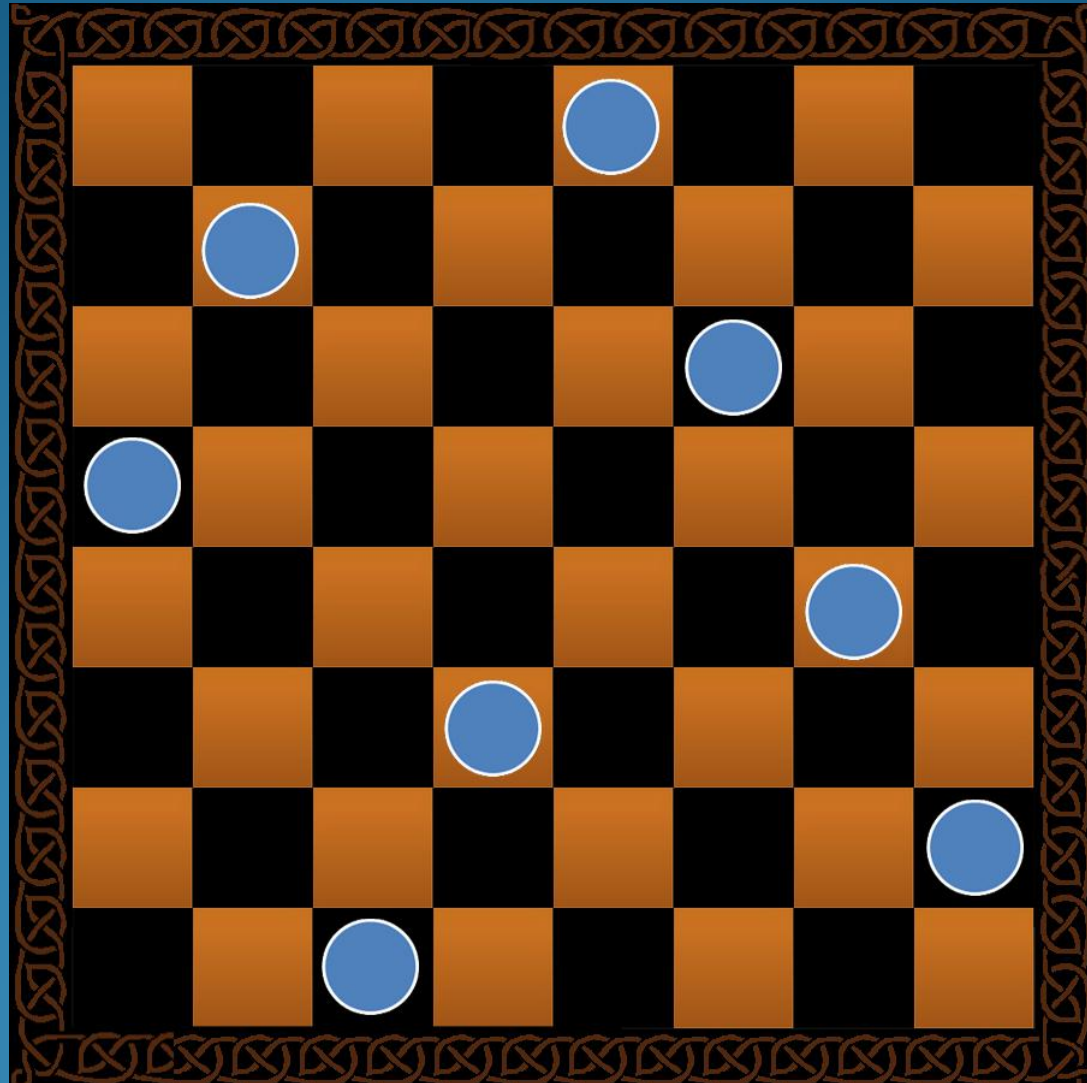
زبان	VC++	۴۵ نمره
زبان	Java	۴۰ نمره
زبان	C#	۳۵ نمره
زبان	C/C++	۳۰ نمره
سایر زبان ها		۲۵ نمره

# پروژه برنامه نویسی ۳

هشت وزیر با الگوریتم سرد و گرم



یک نمونه برنامه هوشمند هشت وزیر ( کامل نشده ) به زبان C# توسط اینجانب





# نمره پروژه هشت وزیر

زبان	VC++	۴۵ نمره
زبان	Java	۴۰ نمره
زبان	C#	۳۵ نمره
زبان	C/C++	۳۰ نمره
سایر زبان ها		۲۵ نمره

# Question ?

# Thank You