

1) سیستمی که مانند انسان رفتار میکند را با ذکر مثال توضیح دهید

تئوری تورینگ، که توسط عالم ریاضیات آلن تورینگ ارائه شده، یک آزمون برای تشخیص هوش مصنوعی است. در این آزمون، یک انسان با یک ماشین و یک انسان دیگر چت می‌کند. اگر انسان نتواند تشخیص دهد کدام یک ماشین است، ماشین قبول آزمون تورینگ شده است. این یعنی ماشین به خوبی رفتار انسان را شبیه‌سازی کرده است. اما تا کنون هیچ سیستم AI به طور قاطعی نتوانسته است این آزمون را پشت سر بگذارد.

2) تفکر عاقلانه چیست و چه آورده ای را در پی خواهد داشت ؟

تفکر عاقلانه به معنای استفاده از منطق و شواهد در فکر کردن است و رفتار عاقلانه. این نوع تفکر می‌تواند به شما کمک کند تا به طور منطقی مسائل را حل کنید، راه‌حل‌های خلاقانه پیدا کنید و تصمیمات بهتری بگیرید. تفکر عاقلانه می‌تواند به گسترش دانش شما کمک کند

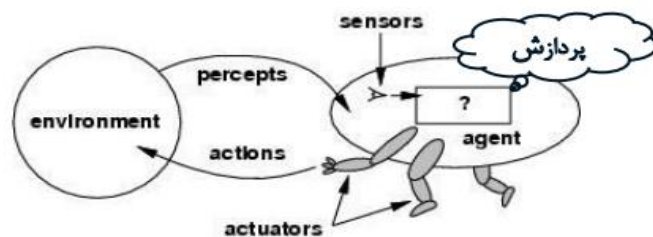
2 مزیت

(اول= عمومیت مبتدی) (دوم = نیازی به تئوری های پیشرفته نیست)

3) اجزای عامل و وظیفه عامل را با رسم شکل و تابع نویسی بررسی کنید

سنسور = وظیفه دریافت مشخصه هایی از محیط

عملگر = وظیفه انجام اعمال بر روی محیط



$$f : P^* \rightarrow A \quad \text{Action} = \text{Agent}(\text{Percept})$$

عمل (ها)

عامل وظیفه دارد رشته دریافت‌های ورودی را به دنباله‌ای از اعمال نگاشت نماید • بنابراین میتوان گفت عامل مانند یک تابع است

عامل میتواند اعمال خود در محیط را درک کند، اما تاثیر آنها بر روی محیط همیشه قابل پیش بینی نیست

4) PEAS برای ربات فوتبالیست و ربات فضانورد ؟

ربات فوتبالیست:

Performer ربات فوتبالیست باید توپ را به دروازه حریف برساند و از گل خوردن جلوگیری کند، در حالی که ربات فضانورد باید Environment زمین فوتبال -چمن- آب و هوا Actuators موتورها و پاها برای حرکت، دست‌ها برای گرفتن و پرتاب توپ Sensors حسگرهای بینایی برای تشخیص موقعیت توپ و دیگر بازیکنان

ربات فضانورد:

Performer مأموریت‌های مختلفی را در فضا انجام دهد از جمله بررسی و تحقیق درباره سیارات، انجام تعمیرات و نگهداری در فضا، و ارسال اطلاعات به زمین.
Environment فضا نوع زمین و خاک
Actuators موتورها و راکت‌ها برای حرکت در فضا، دست‌ها برای انجام کارهای مختلف
Sensors حسگرهای فشار، دما، رادیواسیون و دوربین برای تشخیص و ارسال اطلاعات به سیستم کنترلی.

5) چرا کد پایین با شکست مواجه شده است و راهکار پیشنهادی را نام برده و یکی را توضیح دهید

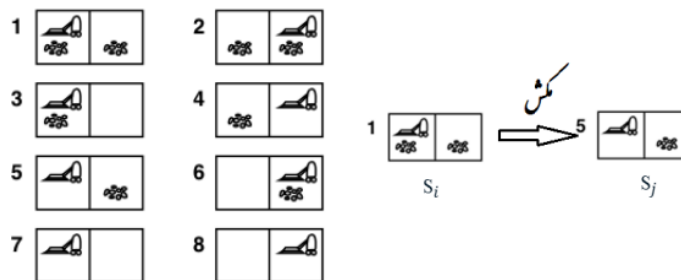
Function TABLE-DRIVEN_AGENT(*percept*) **returns** an action

static: *percepts*, a sequence initially empty
table, a table of actions, indexed by percept sequence

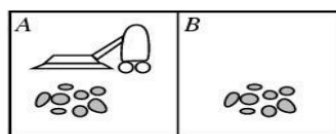
append *percept* to the end of *percepts*
action ← LOOKUP(*percepts*, *table*)
return *action*

در اینجا دقیقاً نوع حرکت را نمیگوید که به چپ یا راست برد یا تک تک حالات چه کارهایی میکنند

(6) دنیای جارو برقی را با توجه به فرمول سازی مساله تشریح کنید



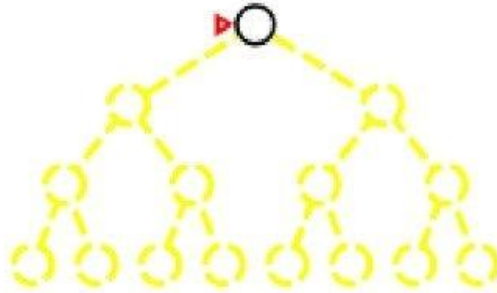
- عامل: یک جارو برقی هوشمند
- محیط: دو اتاق A و B
- سنسور: ۲ دوربین
- دریافت ها: تمیز یا کثیف بودن اتاق، محل استقرار جاروبرقی [A,Clean]
- اعمال: حرکت به چپ، حرکت به راست، مکش و استراحت 😊



(7) جستجوی عمقی را با رسم مرحله به مرحله شرح دهید و در نهایت کارایی الگوریتم را با چهار معیار اندازه گیری بیان کنید.

(۳) جستجوی عمقی

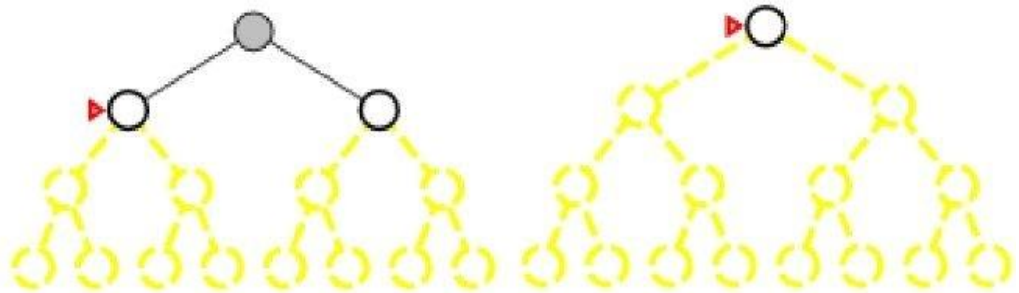
- عمیق ترین گره بسط داده می شود.
- پیاده سازی: *fringe* از طریق یک صف LIFO (پشته)



60

(۳) جستجوی عمقی

- عمیق ترین گره بسط داده می شود.
- پیاده سازی: *fringe* از طریق یک صف LIFO (پشته)



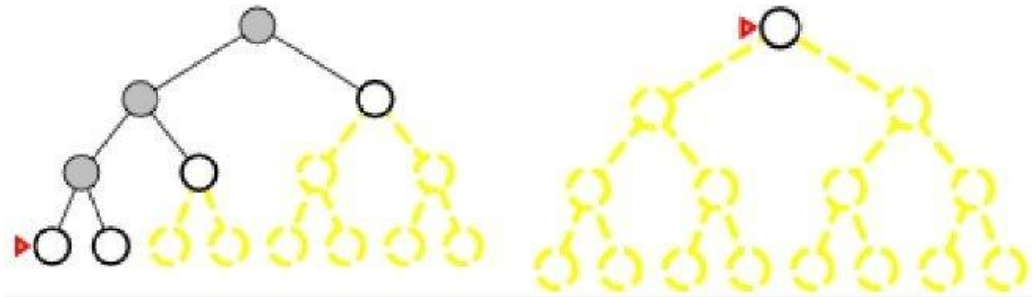
(۳) جستجوی عمقی

- عمیق ترین گره بسط داده می شود.
- پیاده سازی: *fringe* از طریق یک صف LIFO (پشته)



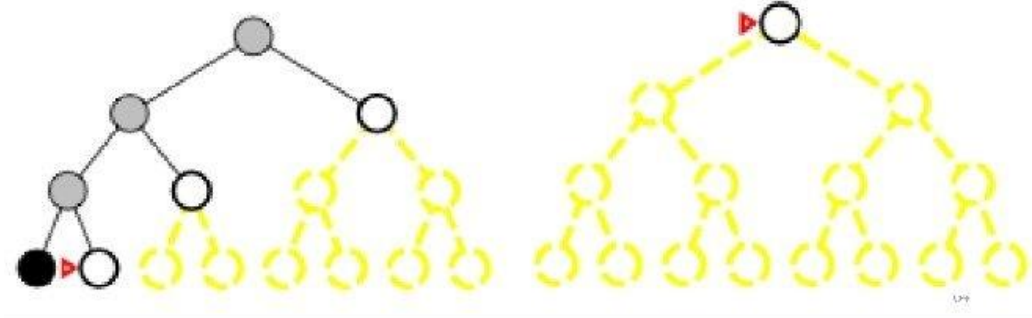
(۳) جستجوی عمقی

- عمیق ترین گره بسط داده می شود.
- پیاده سازی: *fringe* از طریق یک صف LIFO (پشته)



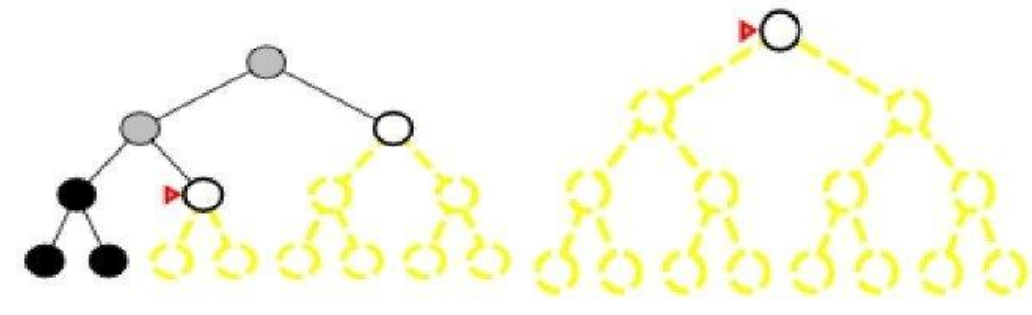
(۳) جستجوی عمقی

- عمیق ترین گره بسط داده می شود.
- پیاده سازی: *fringe* از طریق یک صف LIFO (پشته)



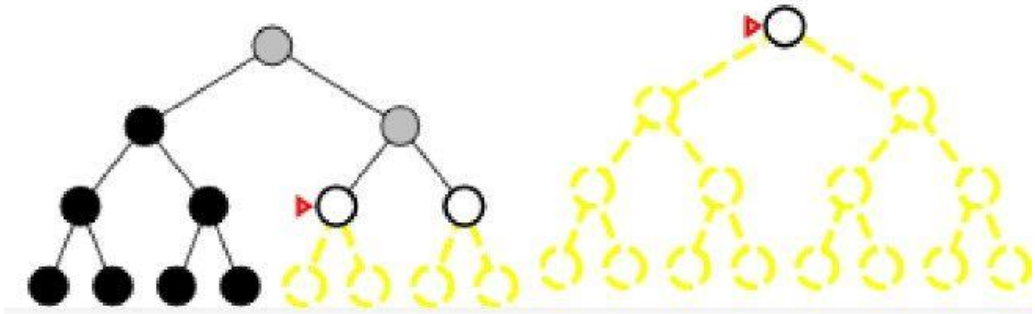
(۳) جستجوی عمقی

- عمیق‌ترین گره بسط داده می‌شود.
- پیاده‌سازی: *fringe* از طریق یک صف LIFO (پشته)



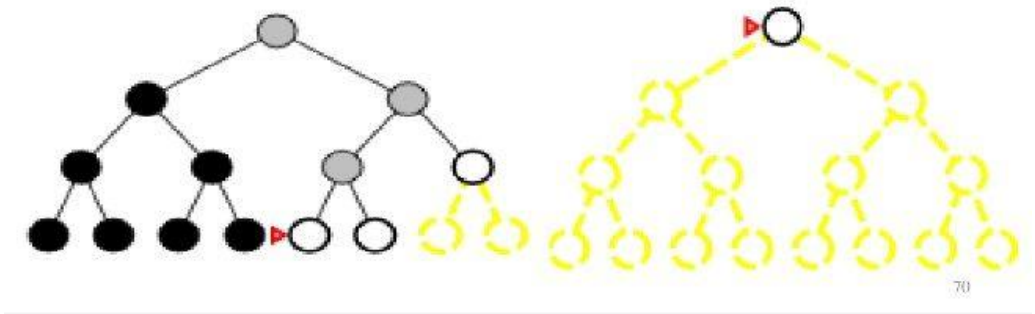
(۳) جستجوی عمقی

- عمیق‌ترین گره بسط داده می‌شود.
- پیاده‌سازی: *fringe* از طریق یک صف LIFO (پشته)



(۳) جستجوی عمقی

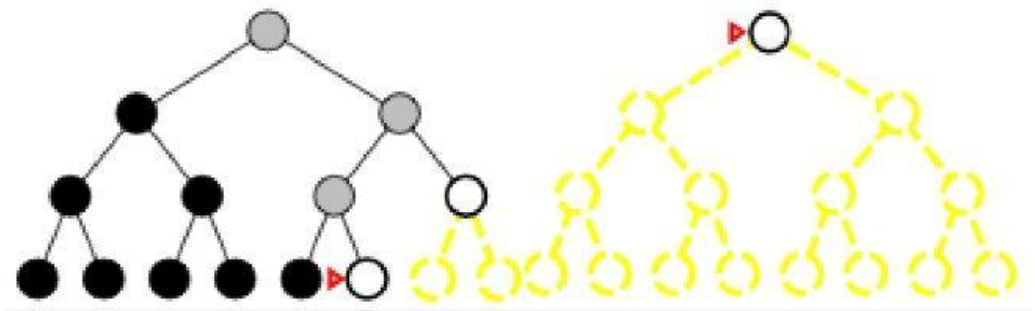
- عمیق‌ترین گره بسط داده می‌شود.
- پیاده‌سازی: *fringe* از طریق یک صف LIFO (پشته)



70

(۳) جستجوی عمقی

- عمیق‌ترین گره بسط داده می‌شود.
- پیاده‌سازی: *fringe* از طریق یک صف LIFO (پشته)



(۳) جستجوی عمقی

- کامل بودن؟ خیر
- ✓ مگر اینکه فضای حالت محدود باشد و حلقه تکرار وجود نداشته باشد.
- بهینه بودن؟ خیر
- ✓ چون کامل نیست.

72

(۳) جستجوی عمقی

- پیچیدگی زمانی؟ $O(b^m)$
- ✓ اگر m خیلی بزرگتر از d باشد به مراتب بدتر است.
- ✓ در بسیاری از مسایل سریعتر از جستجوی BF است.
- پیچیدگی حافظه؟ $O(bm + 1)$
- ✓ در زمان عقبگرد حافظه آزاد می شود.

73

8) برای الگوریتم جستجوی درختی شبکه کد زیر بررسی کنید که استراتژی در کدام از ۴ توابع پیاده سازی شده است؟ توابع را نام برده و عملکرد هر یک را بیان کنید.
راه حل یا اشکال را میگوید:

```
function TREE-SEARCH(problem , [ringe) return a solution or failure
```

کارهای زیر را در حلقه انجام بده:

```
fringe < INSERT(MAKE-NODE(NITIAL-STATE | problem]), fringe)
```

```
loop do
```

در صورتی که خالی بود برگردان، عدم موفقیت را برگردان:

```
if EMPTY?(fringe) then return failure
```

```
node < REMOVE-FIRST(fringe)
```

در این صورت گره را برگردان:

```
if GOAL-TEST[problem] applied to STATE[node] succeeds
```

```
then return SOLUTION(node)
```

```
else
```

این تابع مجموعه ای از گره را برمیگرداند و استراتژی آن اکسپند است:

```
fringe < INSERT-ALL(EXPAND(node, probler), fringe)
```

9) شبکه‌کد زیر مربوط به کدام جستجوی ناآگاهانه می‌باشد و از مزایای کدام جستجوهای دیگر بهره‌برده است با ترسیم شکل توضیح دهید.

```
function DEPTH-LIMITED-SEARCH(problem,limit) return a solution or failure/cutoff
```

```
return RECURSIVE-DLS(MAKE-NODE(NITIAL-STATE[problem]),problem,limit)
```

```
function RECURSIVE-DLS(node, problem, limit) return a solution or failure/cutoff
```

```
cutoff_occurred?<- false
```

```
if GOAL-TEST[problem](STATE[node]) then return SOLUTION(node)
```

```
else if DEPTH[node] == limit then return cutoff
```

```
else for each successor in EXPAND(node, problem) do
```

```
result <-RECURSIVE-DLS(successor, problem, limit)
```

```
if result == cutoff then cutoff_occurred?<- true
```

```
else if result z failure then return result
```

```
if cutoff_occurred?then return cutoffelse return failure
```

(۵) جستجوی عمقی تکراری

■ Limit=0



78

(۵) جستجوی عمقی تکراری

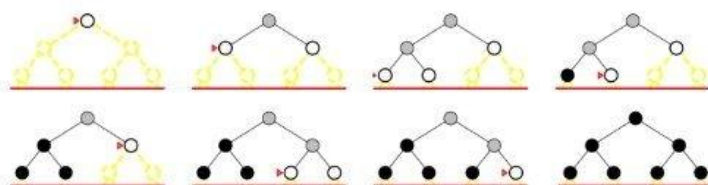
■ Limit=1



79

(۵) جستجوی عمقی تکراری

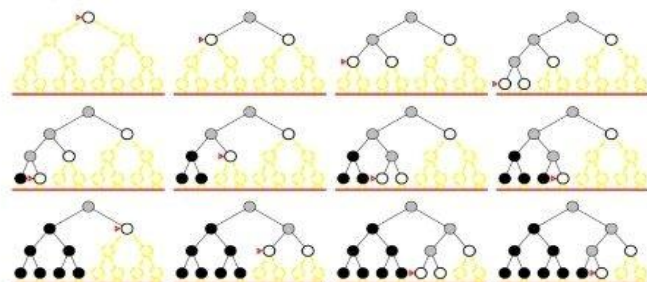
■ Limit=2



80

(۵) جستجوی عمقی تکراری

■ Limit=3



81

10) شش نوع جستجوهای نا آگاهانه جدول زیر را به تفکیک با ۴ معیار مربوطه به اختصار شرح دهید.

- ① جستجوی سطحی - \rightarrow کامل است به شرطی که جواب همیشه در محلی d قابل دسترسی باشد و ماکزیمم انتخاب b محدود باشد / همیشه است به شرطی که مسیر ماکزیمم قابل دسترسی باشد / پیچیدگی زمانی و حافظه $O(b^{d+1})$
- ② جستجوی عمیق - \rightarrow کامل است به شرطی که جواب در عمق d قابل دسترسی باشد و هزینه زمان ماکزیمم در دسترس داشته باشد / همیشه است به شرطی که d محدود باشد / پیچیدگی زمانی و حافظه $O(b^d)$
- ③ جستجوی عمیق - \rightarrow کامل نیست بلکه اشیاء فضای حالت محدود باشد و حلقه تکرار وجود نداشته باشد / همیشه نیست زیرا کامل نیست / پیچیدگی زمانی $O(b^m)$ / پیچیدگی حافظه $O(b^{max})$
- ④ جستجوی عمیق محدود - \rightarrow اگر d محدود باشد غیر کامل است ، اگر d نامحدود باشد اما غیر همیشه است ، اگر d نامحدود باشد و همیشه است / پیچیدگی زمانی $O(b^d)$ / پیچیدگی حافظه $O(b^d)$
- ⑤ جستجوی عمیق تکراری - \rightarrow کامل است به شرطی که حلقه تکرار وجود نداشته باشد و همیشه است اگر هزینه زمان با هم برابر باشد / پیچیدگی زمانی $O(b^d)$ / پیچیدگی حافظه $O(b^d)$
- ⑥ جستجوی دو طرفه - \rightarrow کامل است به شرطی که از جستجوی سطحی استفاده نشود / همیشه است به شرطی که از جستجوی سطحی استفاده نشود / پیچیدگی زمانی و حافظه $O(b^{\frac{d}{2}})$

11) جستجوی A^* را با توجه به جدول $hsld$ با جستجوی حریصانه با رسم درخت به طور کامل توضیح داده و تقارن ها را با دلیل ذکر کنید.

12) الگوریتم زیر را شرح دهید و با توجه به جدول و شکل سوال ، با رسم درخت جستجو توضیح دهید

function Recursive-best-first-search(problem) return a solution or failure
return rfbs (problem, nope(initial-state[problem]),

جواب :

- 1_ بهترین گره و برگ و بهترین جانشین برای آن انتخاب شود
- 2_ اگر مقدار بهترین گره از برگ جانشین آن بیشتر شده ، آنگاه به مسیر جانشین عقبگرد شود

3_ در حین عقبگرد، مقدار $f(t)$ بروزرسانی شود
4_ گره جانشین بسط داده شود

13 چند نوع تابع هیوریستیک را میتوان برای پازل اعداد معرفی کرد با رسم شکل بررسی کنید

جواب :

1_ روند قابل قبول

2_ برای پازل ۸ تایی $h1(m)$ کاشی ها در جای نادرست گذاشته $h2(m)$ مجموع فاصله تا جزیره

14 سه راحل جهت ابداع تابع هیدروبیك نام برده و شرح دهید.

جواب:

۱. از طریق نسخه کوچک تر از مسئله.

۲. از طریق نسخه ساده شده از مسئله.

هر کاشی میتواند به هرجایی منتقل شود.

هر کاشی میتواند به هر خانه همسایه منتقل شود.

هزینه راه حل برای مکعب روبیک را تخمین میزند.

۳. از طریق یادگیری تجربه.

حل مقدار زیادی از مساله (تجربه)

15 انواع جست و جوی محلی را نام برده و ایده هریک را بیان کنید.

جواب:

تپه نوردی: بطور متناوب در جهت بهبود حرکت میکند. زمانی که به قله برسد متوقف میشود. به آینه، گره های برگ قدم

نمیکند، به همین دلیل الگوریتم جست و جوی حریصانه هم معروف است. در صورتیکه بیشتر از یک گره بهترین وجود

داشته باشد، بهترین گره به صورت تصادفی انتخاب میشود.

SA_ اجتناب از گیر کردن در پیشینه های محلی، با اجازه دادن ب انجام حرکت های فرعی (نامناسب) ک حین گذشت زمان

احتمال و تعداد آن کاهش میابد. پیشینه الگوریتم به علوم متالوژی برمیگردد.

پرتوی محلی: از k حالت شروع به جای ۱ حالت شروع بهره میرد، ممکن است الگوریتم از عدم تنوع کافی برخوردار باشد،

تفاوت با تمه با شروع مجدد تصادفی این است که اطلاعات به اشتراک گذاشته میشوند.

ژنتیک

16 الگوریتم زیر را شرح داده و انواع آن را نام برده و بررسی کنید.

Function hill_climbing problem return a state that is a local maximum

Input: problem a problem

Local variables: current a node

Neighbor a node

Current make node initial statele problem loop do

جواب:

به طور متناوب در جهت بهبود حرکت میکند زمانی که به قله میرسد متناوب میشود به آینده گره های برگ توجه نمیکند

به همین دلیل به الگوریتم جست و جوی محلی حریصانه هم مشهور است. در صورتی که بین ۱ گره بهترین وجود داشته

باشد. بهترین گره را بصورت تصادفی انتخاب میکند.

تپه نوردی غیر قطعی: در بین حرکت های رو به بالا یکی به صورت تصادفی انتخاب میشود البته احتمال انتخاب با شیب متناوب است.

تپه نوردی با انتخاب اولین گزینه: گره ها تا حصول یک گره بهتر ربط دادا میشوند.

تپه نوردی تصادفی: از حالت شروع مجدد تصادفی

تا حصول جواب مجددا شروع خواهند نمود.