



Dynamic



عملي مشترك

30/04/2019

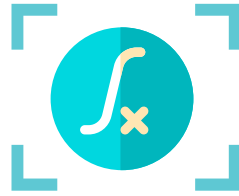
RB Informatics;

مبادئ الذكاء الصناعي

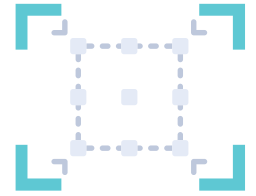
محتويات المحاضرة



تمارين



التوابع المستخدمة للتعامل مع
التخزين الديناميكي



الحقائق الديناميكية

الحقائق الديناميكية

يوجد نوعين من الحقائق في prolog :

1 حقائق ستاتيكية:

male(ahmad) تعرف بشكل صريح ولا يتم تغييرها ضمن البرنامج.

2 حقائق ديناميكية:

تخزن بشكل ديناميكي في الذاكرة.

التوابع المستخدمة للتعامل مع التخزين الديناميكي:

assert()

لتخزين الحقائق

مثال في النافذة الأولى من البرنامج:

```
?- assert(a(1)).  
true.
```

```
?- assert(a(2)).  
true.
```

```
?- assert(a(3)).  
true.
```

```
?- a(X).  
X = 1 ;  
X = 2 ;  
X = 3.
```

retract()

لحذف حقيقة من الذاكرة.

```
?- retract(a(1)).  
true.
```

```
?- a(X).  
X = 2 ;  
X = 3.
```

retractall()

لحذف جميع الحقائق من شكل معين.

```
?- retractall(a(X)).  
true.
```

```
?- a(X).  
false.
```

findall()

findall(fact(X) , a(X) , L):

L=[fact(1) , fact(2) , fact(3)]

تقوم بطباعة جميع الحقائق من الشكل a(X) في السلسلة L.

?- assert(a(1)).

true.

?- assert(a(2)).

true.

?- assert(a(3)).

true.

?- findall(fact(X),a(X),L).

L = [fact(1), fact(2), fact(3)].

write()

?- write(1).

1

true.

للطباعة.

?- write(a).

a

true.

asserta()

لإضافة الحقائق.

?- asserta(a(1)).

true.

?- asserta(a(2)).

true.

الفرق بين asserta() و assert() :

في **assert()** يتم إضافة الحقائق الجديدة في الذاكرة في النهاية.

مثال:

assert(a(1))

assert(a(2))

assert(a(3))

الذاكرة

a(1)
a(2)
a(3)

في **asserta()** تتم إضافة الحقائق الجديدة في الذاكرة في المقدمة (مثل الـ stack).

مثال:

asserta(a(1))

asserta(a(2))

asserta(a(3))

الذاكرة

a(3)
a(2)
a(1)

assertz()

عملها مثل عمل الـ assert().

تمارين مراجعة

1 اكتب برنامج يقوم بطباعة الأرقام من X إلى Y :

get_num (X,X,X) :- ! .

get_num(X,Y,Z) :- X<Y , Z is X ; X1 is X+1 , get_num(X1,Y,Z).

اكتب برنامج يقوم بطباعة مصفوفة من النجوم حسب حيث يكون بُعدها (3,4) :

Printstar(X) :- get_num(1,X,_), write("*"), fail.

Printline(3) :- get_num(1,3,_), nl, printstar(4).

طريقة ثانية لُبُعد غير محدد :

grid_row(1) :- write("*").

grid_row(X) :- X>1, write("*"), X1 is X-1, grid_row(X1).

grid(1,Y) :- grid_row(Y), !.

grid(X,Y) :- X>1, grid_row(Y), nl, X1 is X-1, grid(X1,Y).

