

Logic Programming

Section #7

Ghost

9/12/2009

8 Queens

ده كده الكود والشرح في الصفحة الثانيه ☺

Domains

```
queen=q( integer , integer )
iList = integer*
qList = queen*
```

Predicates

```
nondeterm solution( qList )
nondeterm noattack( queen , qList )
nondeterm member ( integer, iList )
```

Clauses

```
solution([]).
solution([q(X,Y) | Others]):-
    solution(Others),
    member(Y,[1,2,3,4,5,6,7,8]),
    noattack(q(X,Y),Others).

noattack(_,[]).
noattack(q(X,Y),[q(X1,Y1) | Others]):-
    Y <> Y1,
    Y1-Y <> X1-X,
    Y1-Y <> X-X1,
    noattack(q(X,Y),Others).

member( X , [ X | _ ] ).
member( X , [ _ | T ] ):-
    member( X , T ).
```

Goal

```
solution([q(1,Y1),q(2,Y2),q(3,Y3),q(4,Y4),q(5,Y5),q(6,Y6),q(7,Y7),q(8,Y8)]).
```

Domains

queen=q(integer , integer)

هنا عملت Compound Object اسمه Queen اديتلوا 'q' Functor بيشيل 2 Integers.

iList = integer*

عملت List Of Integers اسمها iList.

qList = queen*

عملت list of queens اسمها qList.

Predicates

nondeterm solution(qList)

nondeterm noattack(queen , qList)

nondeterm member (integer, iList)

Clauses

solution([]).

دي كده ال Base Step ان لو ال List اللي جايه فاضيه يبقى هو ده ال Solution لأن فعلاً مش في اي Attack.

solution([q(X,Y) | Others]):-

هنا لو ال List اللي جايه مش فاضيه بقي.

solution(Others),

علشان يطلع ال 8 Queens أول حاجه لازم كل ال Others برضه يبقوا Solution يعني يحصلش اي Attack بين ال Queens.

member(Y,[1,2,3,4,5,6,7,8]),

تاني حاجه ان ال X وال Y لازم يبقوا Integers من 1 لـ 8 وال X أصلاً هتبقى جايه Binded فمش محتاج اعمل عليها Check.

noattack(q(X,Y),Others).

تالت حاجه ان ال Queen اللي هتقطع تبقى Not Attack مع ال Queens الباقيه اللي في ال List.

noattack(_, []).

noattack(q(X,Y),[q(X1,Y1) | Others]) :-

Y <> Y1,

علشان اقول ان ال Queen دي Not Attack مع ال Queens التانيه اول حاجه لازم ال X وال Y يبقوا Different كده ال Queen مش هت Attack Horizontal أو Vertical وبما ان ال X هتبقى جايه Binded فهتبقى جايه Different فعمل Check على ال Y بس.

Y1-Y <> X1-X,

تاني حاجه انهم يكونوا مش على نفس ال +ve Diagonal.

Y1-Y <> X-X1,

تالت حاجه انهم يكونوا مش على نفس ال -ve Diagonal.

noattack(q(X,Y),Others).

هيعمل بقى Backtrack علشان لازم تبقى Not Attack مع كل ال List.

member(X, [_ | _]).

ال X هتبقى Member في ال List لو هيا كانت ال Header ودي ال Base Case.

member(X, [_ | T]) :-

member(X, T).

ال X هتبقى Member في ال List لو كانت Member في ال Tail بتاعت ال List.

Goal

solution([q(1,Y1),q(2,Y2),q(3,Y3),q(4,Y4),q(5,Y5),q(6,Y6),q(7,Y7),q(8,Y8)]).

Assignment:

معاك Code ال 8 Queens وال Game Project اعمل Predicate تعرض حل واحد من حلول ال 8 Queens على ال Grid بتاعت ال Game Project أما ال User يدوس على أي Cell فيها.

Good Luck