

BÀI TẬP 4 - PHÉP TÍNH VỊ TỪ

Thông tin sinh viên

- Họ và tên: Cao Hoài Việt
- MSSV: 22850034

1. Viết vị từ `doublelist(L, R)` với `L` là một danh sách các số được cho, `R` là danh sách `L` với mỗi phần tử được gấp đôi lên.

Bài làm

```
doublelist([], []).
doublelist([H|T], [X|Y]) :-
    X is H*2,
    doublelist(T, Y).
```

2. Viết vị từ `listtran(L, R)` với `L` là một danh sách các số từ 0 đến 10, `R` là danh sách các từ tiếng Anh tương ứng.

Bài làm

```
translate(0, zero).
translate(1, one).
translate(2, two).
translate(3, three).
translate(4, four).
translate(5, five).
translate(6, six).
translate(7, seven).
translate(8, eight).
translate(9, nine).

listtran([], []).
listtran([H|T], [X|Y]) :-
    translate(H, X),
    listtran(T, Y).
```

3. Viết vị từ `sumodd2to(N, S)` với `N` là một số nguyên dương được cho, `S` sẽ là tổng bình phương các số lẻ từ 1 đến `N` (nghĩa là, $S = 1^2 + 3^2 + 5^2 + \dots + X^2$, `X` là `N` nếu `N` lẻ và là `N-1` nếu `N` chẵn).

Bài làm

```
is_even(N) :-
    0 ::= N mod 2.

sum_on_list([], 0).
sum_on_list([H|T], S) :-
    sum_on_list(T, R),
    S is H * H + R.

sumodd2to(0, 0).
sumodd2to(X, S) :-
    numlist(1, X, L),
    exclude(is_even, L, R),
    sum_on_list(R, S).
```

4. Viết vị từ `removeNeg(L, R)` với `L` là một danh sách các số được cho, `R` là danh sách `L` sau khi xóa đi các số âm.

Bài làm cách 1. Tư định nghĩa KB

```
removeNeg([], []).
removeNeg([H|T], L) :-
    H < 0,
    removeNeg(T, L).
removeNeg([H|T], [H|L]) :-
    H >= 0,
    removeNeg(T, L).
```

Bài làm cách 2. Sử dụng hàm có sẵn của Prolog

```
is_negative(X) :-
    X < 0.
removeNeg(L, R) :-
    exclude(is_negative, L, R).
```

5. Viết vị từ minmax(L, X, Y) tìm phần tử nhỏ nhất X và lớn nhất Y trong danh sách các số nguyên L.

Bài làm cách 1. Tư định nghĩa KB

```
min([], []).
min([L], L).
min([H,K|T], R) :-
    H =< K,
    min([H|T], R).
min([H,K|T], R) :-
    H > K,
    min([K|T], R).
```

```
max([], []).
max([L], L).
max([H,K|T], R) :-
    H > K,
    max([H|T], R).
max([H,K|T], R) :-
    H =< K,
    max([K|T], R).
```

```
minmax([], [], []).
minmax(L, X, Y) :-
    min(L, X),
    max(L, Y).
```

Bài làm cách 2. Sử dụng hàm có sẵn của Prolog

```
minmax(L, X, Y) :-
    min_list(L, X),
    max_list(L, Y).
```