AI LAB 03

transport2.pl

File Edit Browse Compile Prolog Pce Help

transport2.pl

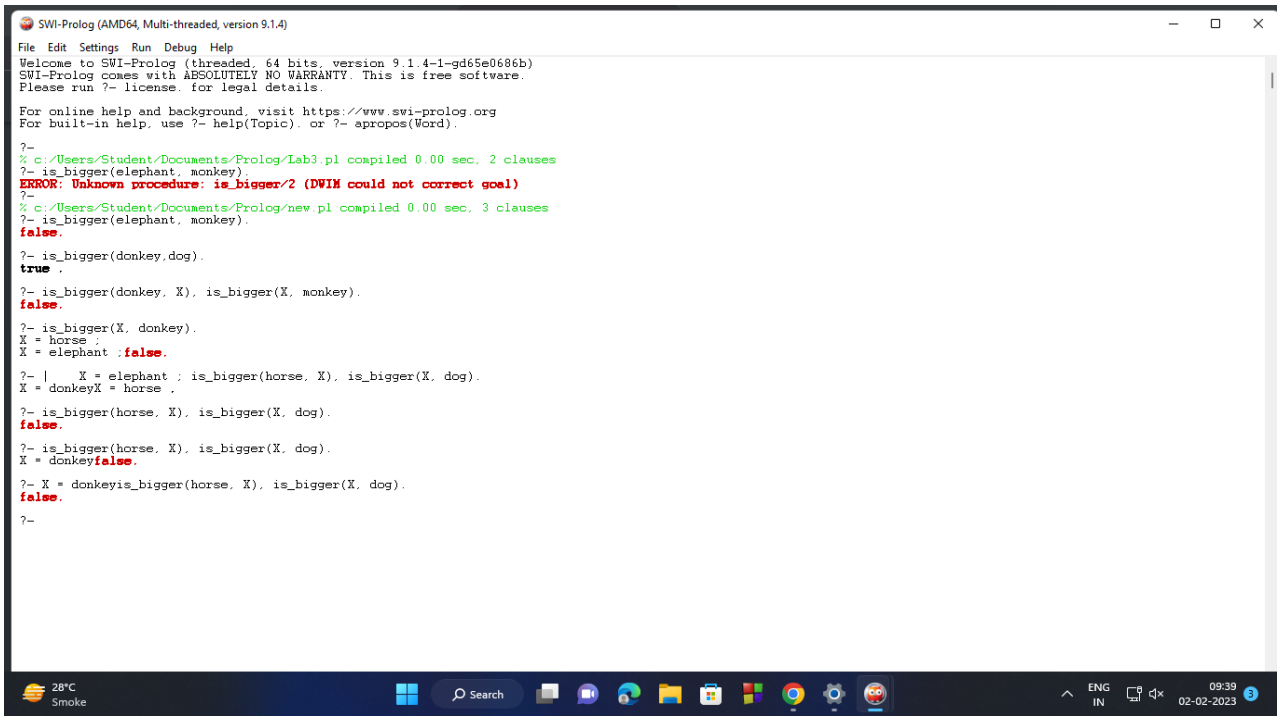
```
% Define the cities and the costs to travel between them
city(1, 2, 10).
city(2, 3, 20).
city(3, 4, 30).
city(3, 5, 20).
city(4, 5, 40).
city(2, 4, 20).

% Define the recursive predicate to find the minimum cost path
find_path(Start, Goal, Path, Cost) :-
    travel(Start, Goal, [Start], 0, Path, Cost).

% Define the predicate to traverse the cities and keep track of the path and cost
travel(Goal, Goal, Path, Cost, Path, Cost).
travel(Current, Goal, Path, Cost, FinalPath, FinalCost) :-
    city(Current, Next, CostToNext),
    \+ member(Next, Path),
    NewCost is Cost + CostToNext,
    travel(Next, Goal, [Next|Path], NewCost, FinalPath, FinalCost).

% Find the minimum cost path
find_min_cost(Start, Goal, Path, Cost) :-
    setof((P, C), find_path(Start, Goal, P, C), Set),
    Set = [(_, Cost)|_],
    find_path(Start, Goal, Path, Cost).

% Call the predicate to find the minimum cost path from the first city to the goal city
main :-
    find_min_cost(1, 5, Path, Cost),
    write('Path: '),
    write(Path),
    write('\nCost: '),
    write(Cost),
    nl.
```

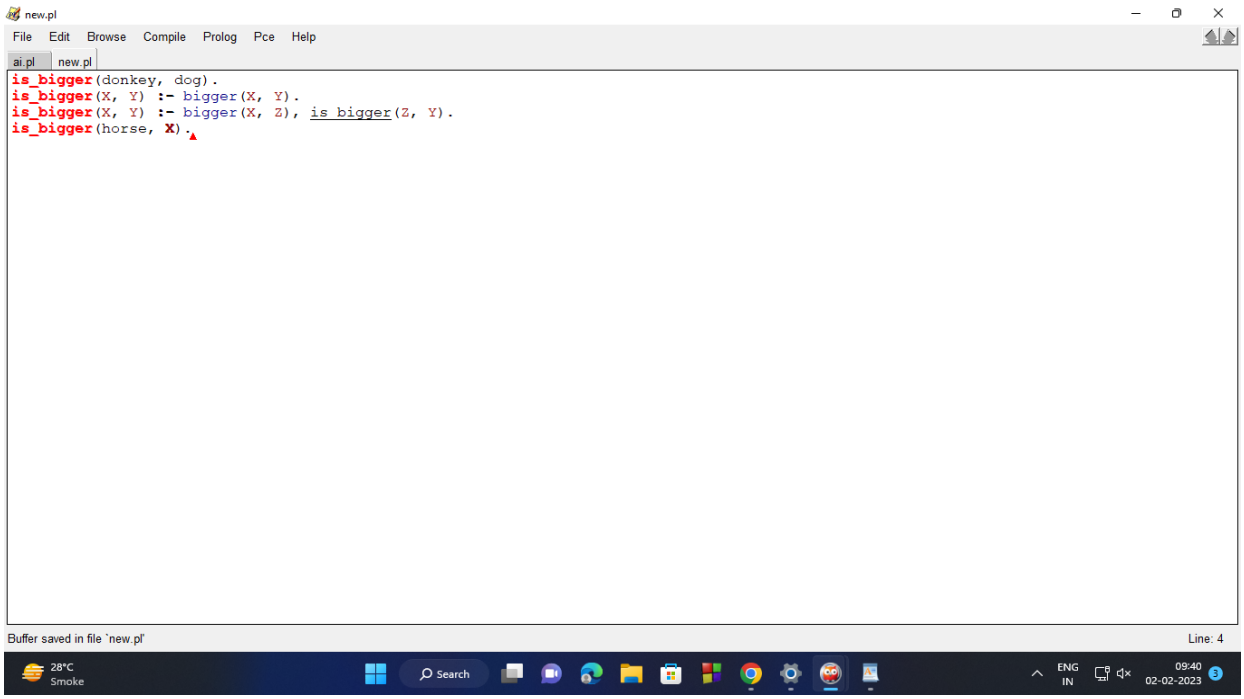
comment(line) Line: 1

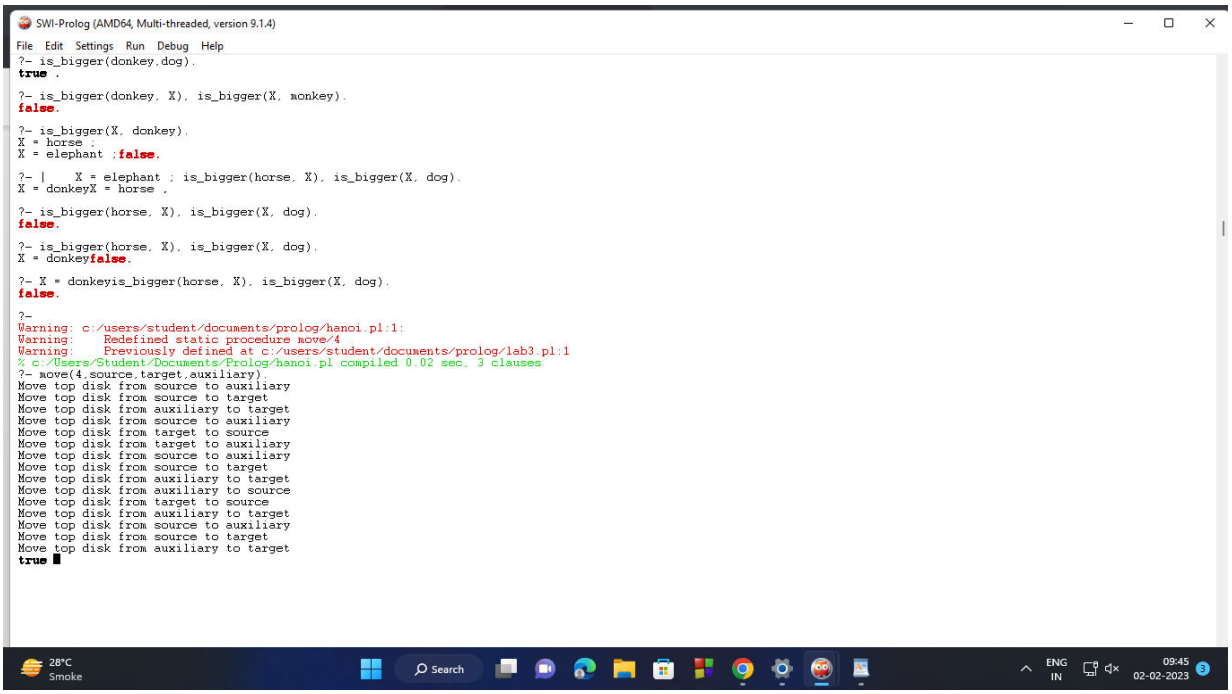
SWI-Prolog (AMD64, Multi-threaded, version 9.0.4)

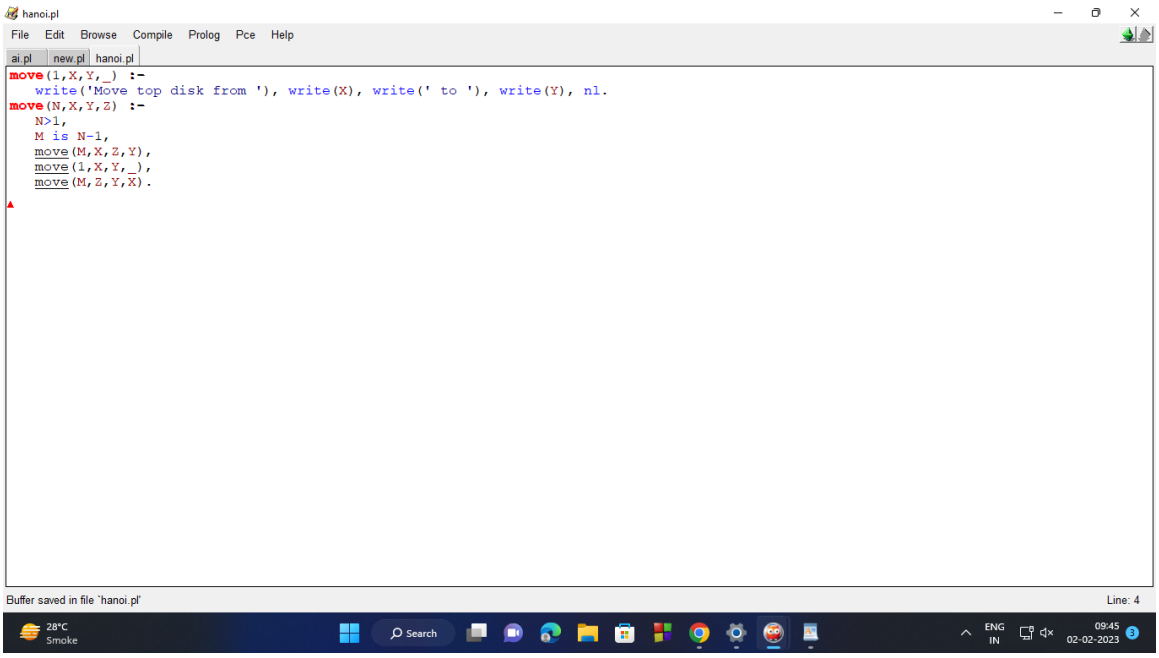
File Edit Settings Run Debug Help

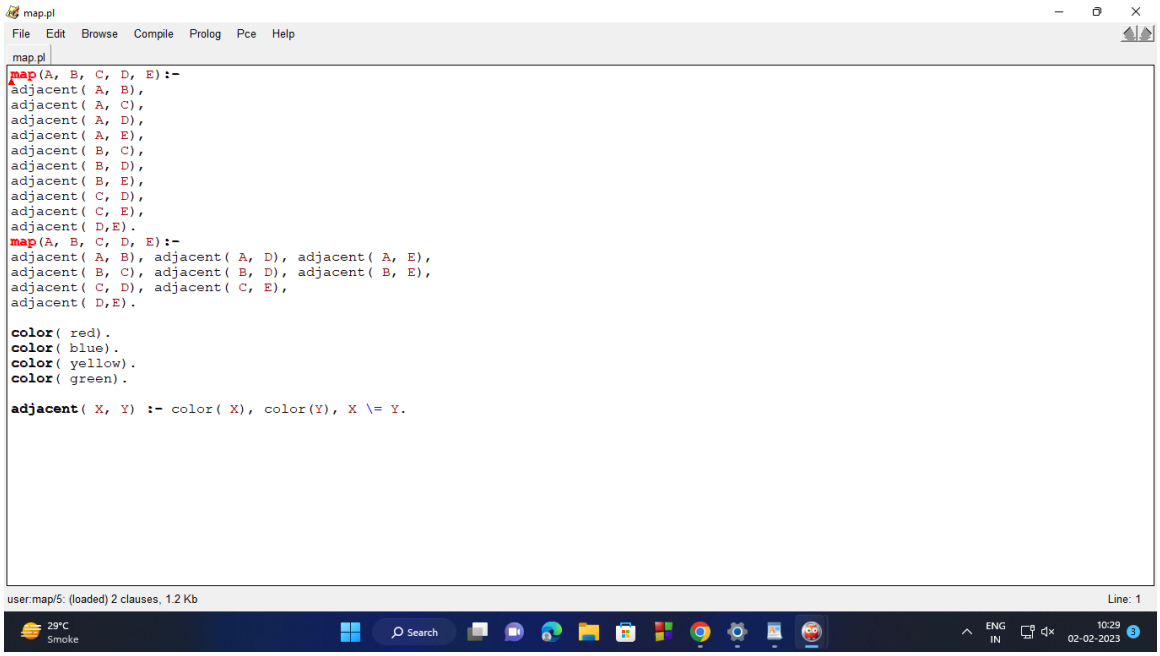
```
Welcome to SWI-Prolog (threaded, 64 bits, version 9.0.4)
SWI-Prolog comes with ABSOLUTELY NO WARRANTY. This is free software.
Please run ?- license. for legal details.

For online help and background, visit https://www.swi-prolog.org
For built-in help, use ?- help(Topic). or ?- apropos(Word).

?-
% c:/Users/aditi/OneDrive/Documents/Prolog/transport2.pl compiled 0.00 sec, 11 clauses
?-
|    main().
Path: [5,3,2,1]
Cost: 50
true
```

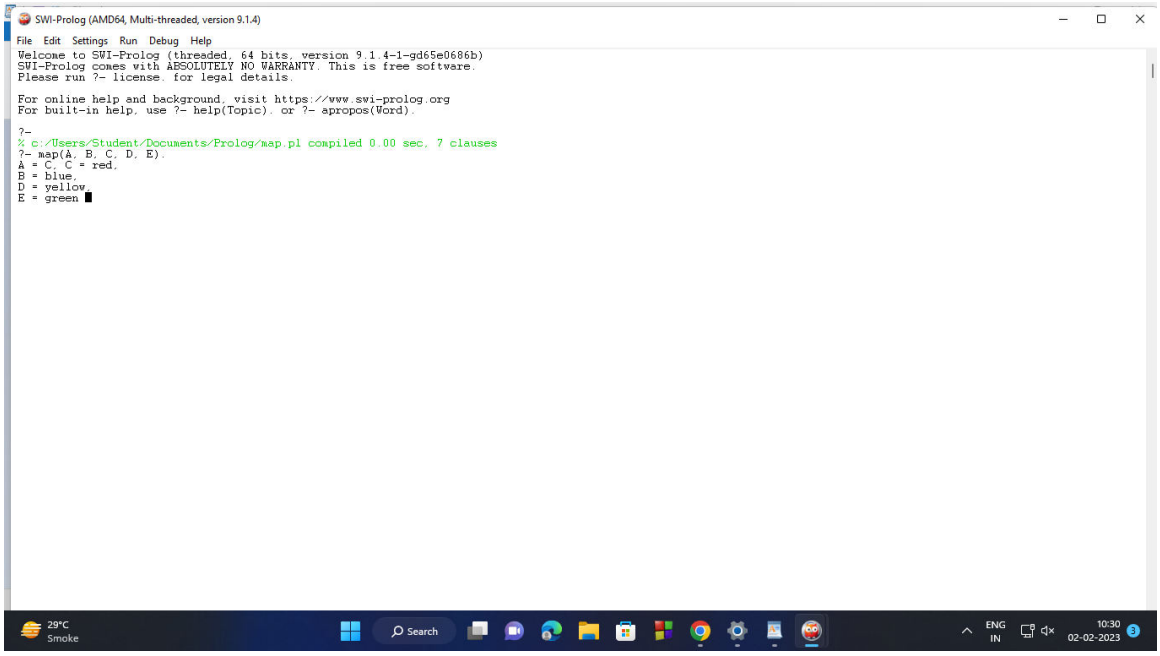

SWI-Prolog (AMD64, Multi-threaded, version 9.1.4)
File Edit Settings Run Debug Help
Welcome to SWI-Prolog (threaded, 64 bits, version 9.1.4-1-gd65e0686b)
SWI-Prolog comes with ABSOLUTELY NO WARRANTY. This is free software.
Please run ?- license. for legal details.
For online help and background, visit https://www.swi-prolog.org
For built-in help, use ?- help(Topic). or ?- apropos(Word).
?-
% c:/Users/Student/Documents/Prolog/Lab3.pl compiled 0.00 sec, 2 clauses
?- is_bigger(elephant, monkey).
ERROR: Unknown procedure: is_bigger/2 (DWIM could not correct goal)
?-
% c:/Users/Student/Documents/Prolog/new.pl compiled 0.00 sec, 3 clauses
?- is_bigger(elephant, monkey).
false.
?- is_bigger(donkey,dog).
true .
?- is_bigger(donkey, X), is_bigger(X, monkey).
false.
?- is_bigger(X, donkey).
X = horse ;
X = elephant ;false.
?- | X = elephant ; is_bigger(horse, X), is_bigger(X, dog).
X = donkeyX = horse ,
?- is_bigger(horse, X), is_bigger(X, dog).
false.
?- is_bigger(horse, X), is_bigger(X, dog).
X = donkeyfalse.
?- X = donkeyis_bigger(horse, X), is_bigger(X, dog).
false.
?-
28°C
Smoke
Search
ENG
IN
09:39
02-02-2023

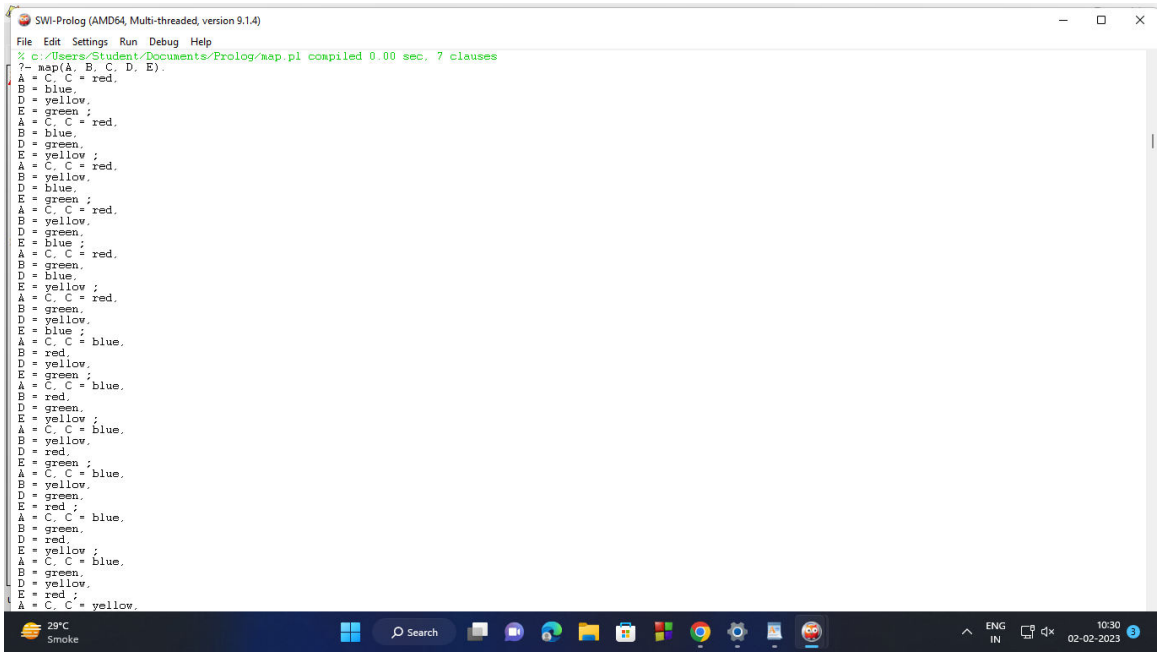

new.pl
File Edit Browse Compile Prolog Pce Help
ai.pl
new.pl
is_bigger(donkey, dog).
is_bigger(X, Y) :- bigger(X, Y).
is_bigger(X, Y) :- bigger(X, Z), is_bigger(Z, Y).
is_bigger(horse, X).
Buffer saved in file `new.pl'
Line: 4
28°C
Smoke
Search
ENG
IN
09:40
02-02-2023


SWI-Prolog (AMD64, Multi-threaded, version 9.1.4)
File Edit Settings Run Debug Help
?- is_bigger(donkey,dog).
true .
?- is_bigger(donkey, X), is_bigger(X, monkey).
false.
?- is_bigger(X, donkey).
X = horse ;
X = elephant ;false.
?- | X = elephant ; is_bigger(horse, X), is_bigger(X, dog).
X = donkeyX = horse .
?- is_bigger(horse, X), is_bigger(X, dog).
false.
?- is_bigger(horse, X), is_bigger(X, dog).
X = donkeyfalse.
?- X = donkeyis_bigger(horse, X), is_bigger(X, dog).
false.
?-
Warning: c:/users/student/documents/prolog/hanoi.pl:1:
Warning: Redefined static procedure move/4
Warning: Previously defined at c:/users/student/documents/prolog/lab3.pl:1
% c:/Users/Student/Documents/Prolog/hanoi.pl compiled 0.02 sec, 3 clauses
?- move(4,source,target,auxiliary).
Move top disk from source to auxiliary
Move top disk from source to target
Move top disk from auxiliary to target
Move top disk from source to auxiliary
Move top disk from target to source
Move top disk from target to auxiliary
Move top disk from source to auxiliary
Move top disk from source to target
Move top disk from auxiliary to target
Move top disk from auxiliary to source
Move top disk from target to source
Move top disk from auxiliary to target
Move top disk from source to auxiliary
Move top disk from source to target
Move top disk from auxiliary to target
true
28°C Smoke
Search
ENG IN
09:45 02-02-2023


hanoi.pl
File Edit Browse Compile Prolog Pce Help
ai.pl new.pl hanoi.pl
move(1,X,Y,_) :-
write('Move top disk from '), write(X), write(' to '), write(Y), nl.
move(N,X,Y,Z) :-
N>1,
M is N-1,
move(M,X,Z,Y),
move(1,X,Y,_),
move(M,Z,Y,X).
Buffer saved in file 'hanoi.pl'
Line: 4
28°C Smoke
Search
ENG IN
09:45 02-02-2023


map.pl
File Edit Browse Compile Prolog Pce Help
map.pl
map(A, B, C, D, E):-
adjacent(A, B),
adjacent(A, C),
adjacent(A, D),
adjacent(A, E),
adjacent(B, C),
adjacent(B, D),
adjacent(B, E),
adjacent(C, D),
adjacent(C, E),
adjacent(D,E).
map(A, B, C, D, E):-
adjacent(A, B), adjacent(A, D), adjacent(A, E),
adjacent(B, C), adjacent(B, D), adjacent(B, E),
adjacent(C, D), adjacent(C, E),
adjacent(D,E).
color(red).
color(blue).
color(yellow).
color(green).
adjacent(X, Y) :- color(X), color(Y), X \= Y.
user:map/5: (loaded) 2 clauses, 1.2 Kb
Line: 1
29°C
Smoke
Search
ENG
IN
10:29
02-02-2023


SWI-Prolog (AMD64, Multi-threaded, version 9.1.4)
File Edit Settings Run Debug Help
Welcome to SWI-Prolog (threaded, 64 bits, version 9.1.4-1-gd65e0686b)
SWI-Prolog comes with ABSOLUTELY NO WARRANTY. This is free software.
Please run ?- license. for legal details.
For online help and background, visit https://www.swi-prolog.org
For built-in help, use ?- help(Topic). or ?- apropos(Word).
?-
% c:/Users/Student/Documents/Prolog/map.pl compiled 0.00 sec, 7 clauses
?- map(A, B, C, D, E).
A = C, C = red,
B = blue,
D = yellow,
E = green
29°C
Smoke
Search
ENG
IN
10:30
02-02-2023


SWI-Prolog (AMD64, Multi-threaded, version 9.1.4)
File Edit Settings Run Debug Help
% c:/Users/Student/Documents/Prolog/map.pl compiled 0.00 sec, 7 clauses
?- map(A, B, C, D, E).
A = C, C = red,
B = blue,
D = yellow,
E = green ;
A = C, C = red,
B = blue,
D = green,
E = yellow ;
A = C, C = red,
B = yellow,
D = blue,
E = green ;
A = C, C = red,
B = yellow,
D = green,
E = blue ;
A = C, C = red,
B = green,
D = blue,
E = yellow ;
A = C, C = red,
B = green,
D = yellow,
E = blue ;
A = C, C = blue,
B = red,
D = yellow,
E = green ;
A = C, C = blue,
B = red,
D = green,
E = yellow ;
A = C, C = blue,
B = yellow,
D = red,
E = green ;
A = C, C = blue,
B = yellow,
D = green,
E = red ;
A = C, C = blue,
B = green,
D = red,
E = yellow ;
A = C, C = blue,
B = green,
D = yellow,
E = red ;
A = C, C = yellow,
29°C
Smoke
Search
ENG
IN
10:30
02-02-2023