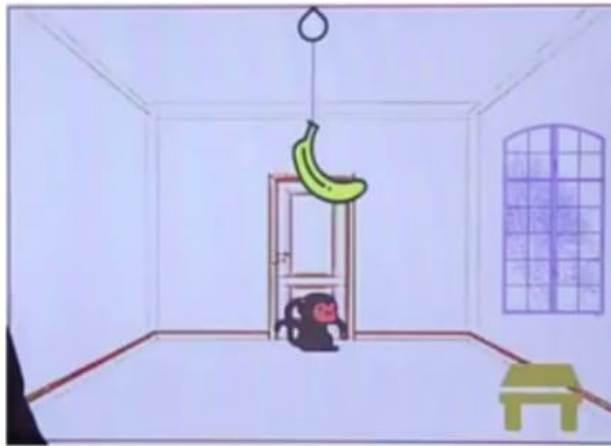


Assignment – 6

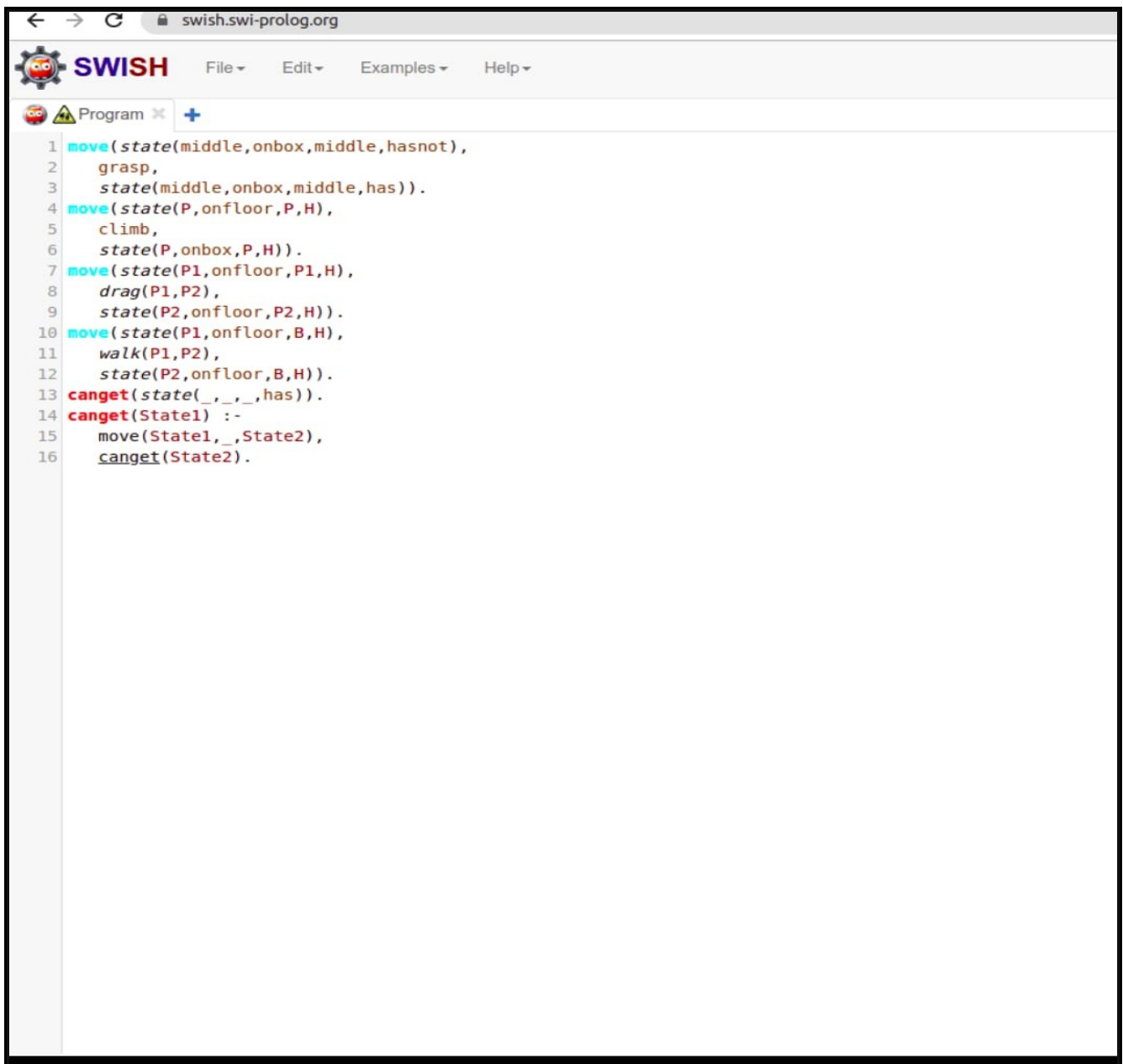
Assignment 6

Monkey is on the floor, at door. A block is on floor, at window. Banana is hanging from the roof at the middle of room. Problem is "How monkey can get the banana?"
Write a PROLOG program for Monkey Banana Problem.



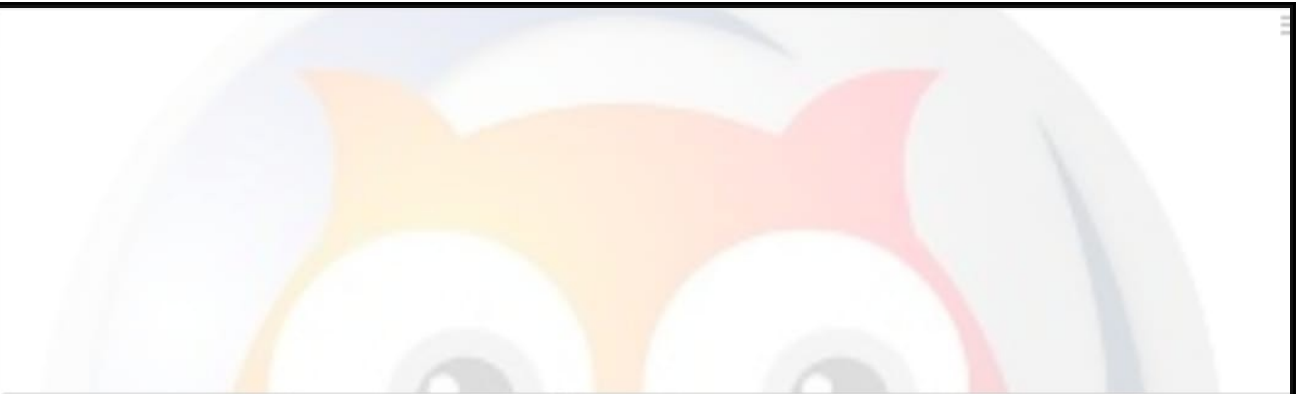
```
move(state(middle,onbox,middle,hasnot),
    grasp,
    state(middle,onbox,middle,has)).
move(state(P,onfloor,P,H),
    climb,
    state(P,onbox,P,H)).
move(state(P1,onfloor,P1,H),
    drag(P1,P2),
```


```
state(P2,onfloor,P2,H)).
move(state(P1,onfloor,B,H),
walk(P1,P2),
state(P2,onfloor,B,H)).
canget(state(_,_,_,has)).
canget(State1) :-
move(State1,_,State2),
canget(State2).
```




The screenshot shows the SWISH Prolog editor interface. The browser address bar displays "swish.swi-prolog.org". The interface includes a menu bar with "File", "Edit", "Examples", and "Help". Below the menu bar is a tab labeled "Program" with a plus icon to add more programs. The main editing area contains a Prolog program with 16 lines of code, which is a slightly modified version of the code shown in the first block. The code is as follows:

```
1 move(state(middle,onbox,middle,hasnot),
2   grasp,
3   state(middle,onbox,middle,has)).
4 move(state(P,onfloor,P,H),
5   climb,
6   state(P,onbox,P,H)).
7 move(state(P1,onfloor,P1,H),
8   drag(P1,P2),
9   state(P2,onfloor,P2,H)).
10 move(state(P1,onfloor,B,H),
11   walk(P1,P2),
12   state(P2,onfloor,B,H)).
13 canget(state(_,_,_,has)).
14 canget(State1) :-
15   move(State1,_,State2),
16   canget(State2).
```



 trace

true

 `canget(state(atdoor, onfloor, atwindow, hasnot)).`

true


Next

10


100

1,000


Stop

 true

true

 trace

true

 `canget(state(atdoor, onfloor, atwindow, hasnot))`

true

Next

10

100

1,000

Stop

?- `canget(state(atdoor, onfloor, atwindow, hasnot))`

Examples▲

History▲

Solutions▲

☐ table results

Run