Name: Brijesh Rohit

Admission no.: U19CS009

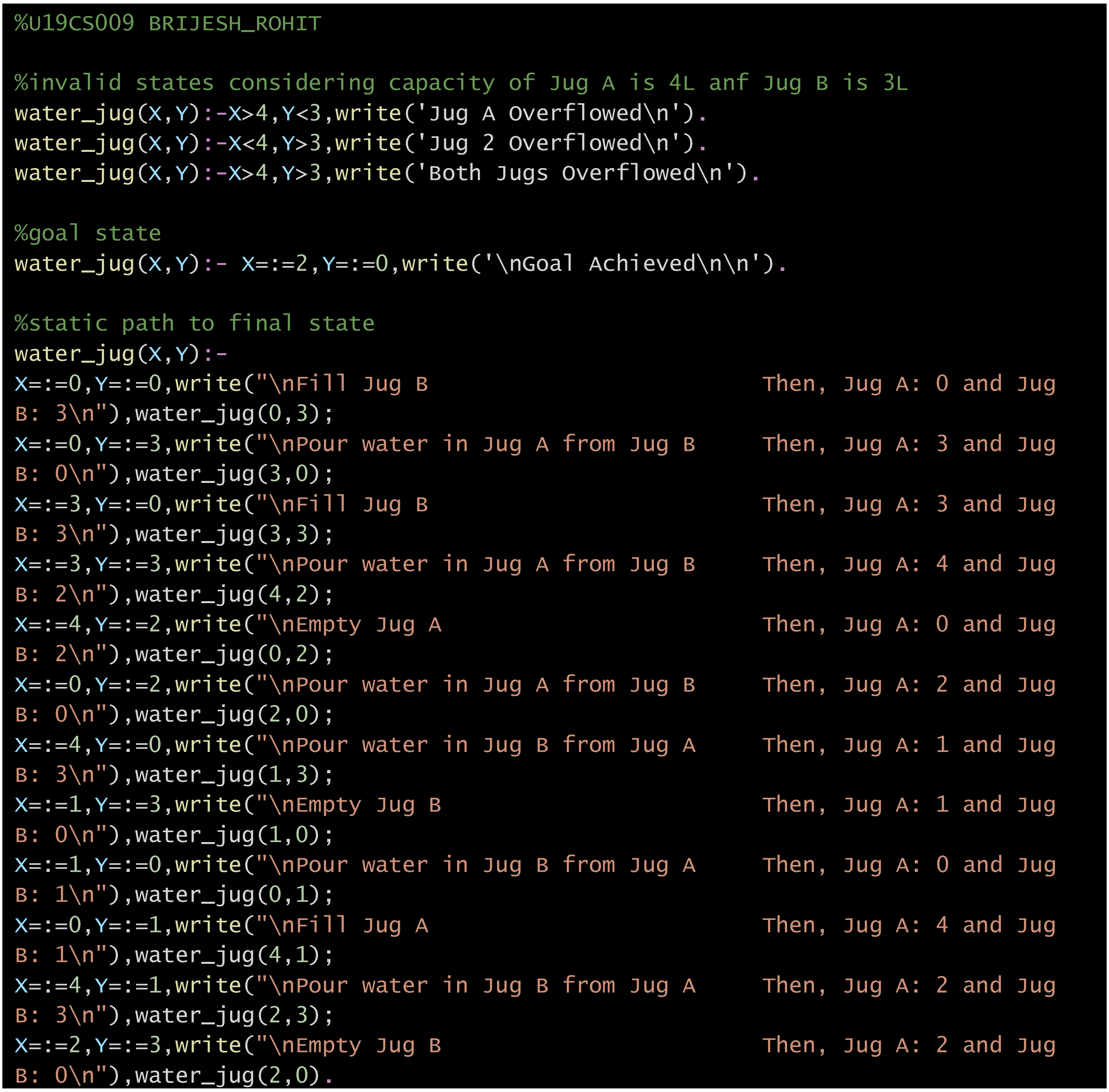
ARTIFICIAL INTELLIGENCE - ASSIGNMENT - 1

Design State Space Search for Water Jug Problem and Implement same problem in prolog.

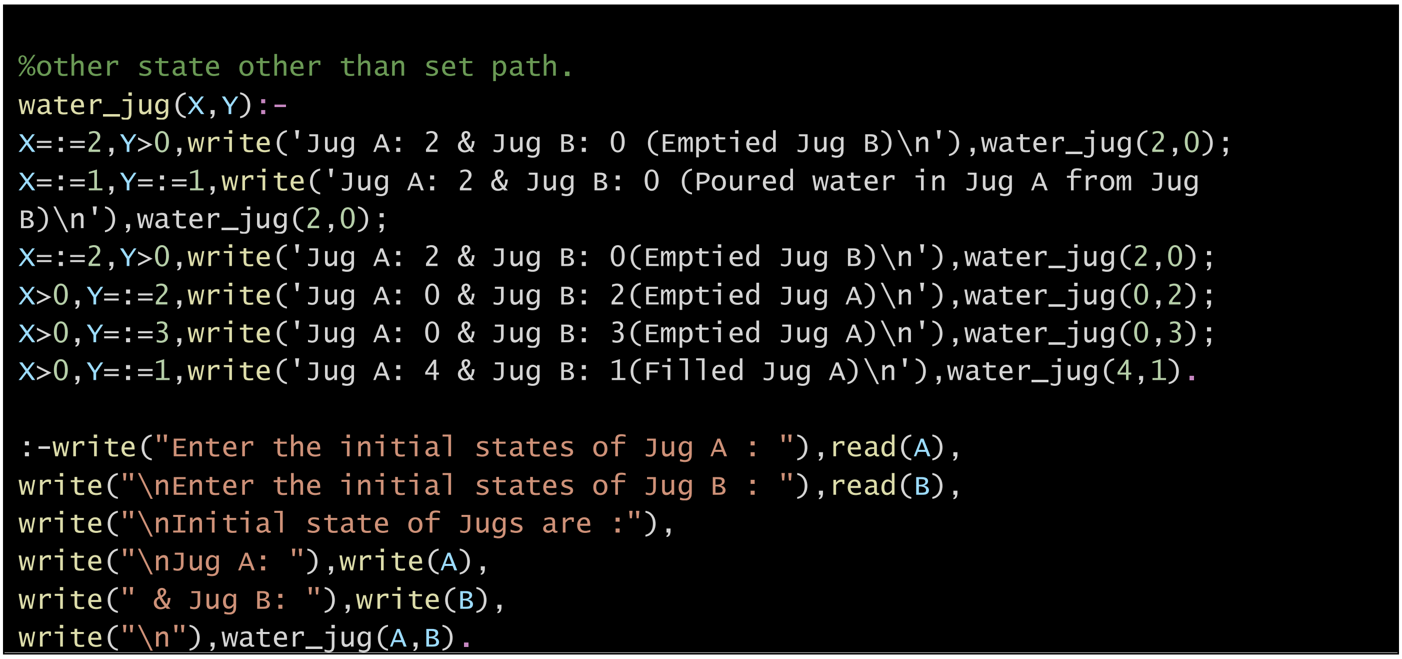
Problem: There are two jugs of volume A litre and B litre. Neither has any measuring mark on it.There is a pump that can be used to fill the jugs with water.How can you get exactly of water into the Assuming that we have unlimited supply of water.

For e.g., one having the capacity to hold 3 gallons of water and the other has the capacity to hold 4 gallons of water. There is no other measuring equipment available and the jugs also do not have any kind of marking on them. So, the agent's task here is to fill the 4-gallon jug with 2 gallons of water by using only these two jugs and no other material. Initially, both our jugs are empty.

CODE=>



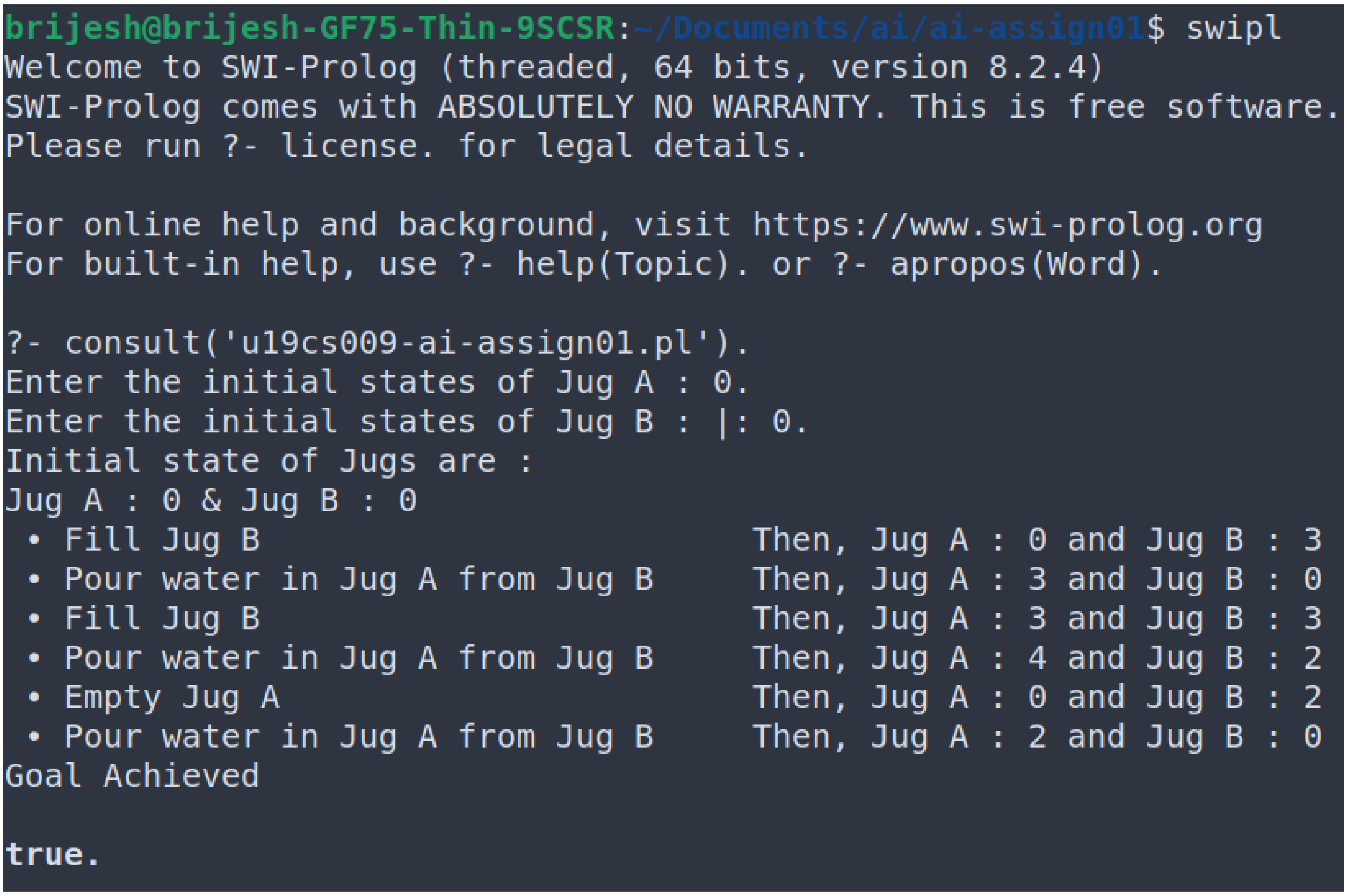
```
%U19CS009 BRIJESH_ROHIT

%invalid states considering capacity of Jug A is 4L anf Jug B is 3L
water_jug(X,Y):-X>4,Y<3,write('Jug A Overflowed\n').
water_jug(X,Y):-X<4,Y>3,write('Jug 2 Overflowed\n').
water_jug(X,Y):-X>4,Y>3,write('Both Jugs Overflowed\n').

%goal state
water_jug(X,Y):- X=:=2,Y=:=0,write('\nGoal Achieved\n\n').

%static path to final state
water_jug(X,Y):-
X=:=0,Y=:=0,write("\nFill Jug B                       Then, Jug A: 0 and Jug
B: 3\n"),water_jug(0,3);
X=:=0,Y=:=3,write("\nPour water in Jug A from Jug B     Then, Jug A: 3 and Jug
B: 0\n"),water_jug(3,0);
X=:=3,Y=:=0,write("\nFill Jug B                       Then, Jug A: 3 and Jug
B: 3\n"),water_jug(3,3);
X=:=3,Y=:=3,write("\nPour water in Jug A from Jug B     Then, Jug A: 4 and Jug
B: 2\n"),water_jug(4,2);
X=:=4,Y=:=2,write("\nEmpty Jug A                      Then, Jug A: 0 and Jug
B: 2\n"),water_jug(0,2);
X=:=0,Y=:=2,write("\nPour water in Jug A from Jug B     Then, Jug A: 2 and Jug
B: 0\n"),water_jug(2,0);
X=:=4,Y=:=0,write("\nPour water in Jug B from Jug A     Then, Jug A: 1 and Jug
B: 3\n"),water_jug(1,3);
X=:=1,Y=:=3,write("\nEmpty Jug B                      Then, Jug A: 1 and Jug
B: 0\n"),water_jug(1,0);
X=:=1,Y=:=0,write("\nPour water in Jug B from Jug A     Then, Jug A: 0 and Jug
B: 1\n"),water_jug(0,1);
X=:=0,Y=:=1,write("\nFill Jug A                       Then, Jug A: 4 and Jug
B: 1\n"),water_jug(4,1);
X=:=4,Y=:=1,write("\nPour water in Jug B from Jug A     Then, Jug A: 2 and Jug
B: 3\n"),water_jug(2,3);
X=:=2,Y=:=3,write("\nEmpty Jug B                      Then, Jug A: 2 and Jug
B: 0\n"),water_jug(2,0).
```



```
%other state other than set path.
water_jug(X,Y):-
X=:=2,Y>0,write('Jug A: 2 & Jug B: 0 (Emptied Jug B)\n'),water_jug(2,0);
X=:=1,Y=:=1,write('Jug A: 2 & Jug B: 0 (Poured water in Jug A from Jug
B)\n'),water_jug(2,0);
X=:=2,Y>0,write('Jug A: 2 & Jug B: 0(Emptied Jug B)\n'),water_jug(2,0);
X>0,Y=:=2,write('Jug A: 0 & Jug B: 2(Emptied Jug A)\n'),water_jug(0,2);
X>0,Y=:=3,write('Jug A: 0 & Jug B: 3(Emptied Jug A)\n'),water_jug(0,3);
X>0,Y=:=1,write('Jug A: 4 & Jug B: 1(Filled Jug A)\n'),water_jug(4,1).

:-write("Enter the initial states of Jug A : "),read(A),
write("\nEnter the initial states of Jug B : "),read(B),
write("\nInitial state of Jugs are :"),
write("\nJug A: "),write(A),
write(" & Jug B: "),write(B),
write("\n"),water_jug(A,B).
```

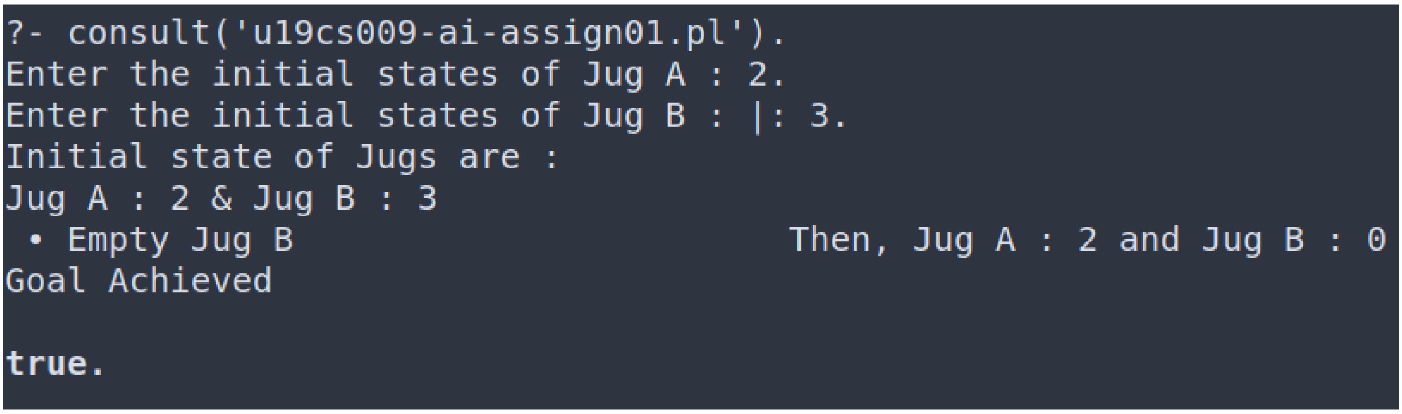
OUTPUT=>

(0,0)



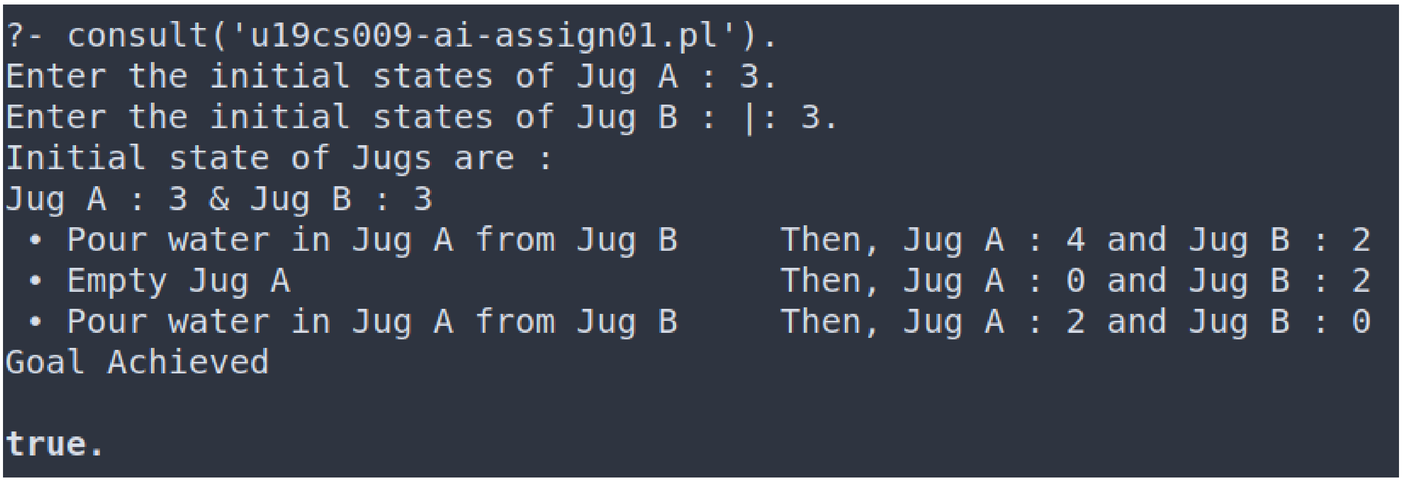
```
brijesh@brijesh-GF75-Thin-9SCSR:~/Documents/ai/ai-assign01$ swipl
Welcome to SWI-Prolog (threaded, 64 bits, version 8.2.4)
SWI-Prolog comes with ABSOLUTELY NO WARRANTY. This is free software.
Please run ?- license. for legal details.

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

?- consult('u19cs009-ai-assign01.pl').
Enter the initial states of Jug A : 0.
Enter the initial states of Jug B : |: 0.
Initial state of Jugs are :
Jug A : 0 & Jug B : 0
 • Fill Jug B                        Then, Jug A : 0 and Jug B : 3
 • Pour water in Jug A from Jug B    Then, Jug A : 3 and Jug B : 0
 • Fill Jug B                        Then, Jug A : 3 and Jug B : 3
 • Pour water in Jug A from Jug B    Then, Jug A : 4 and Jug B : 2
 • Empty Jug A                       Then, Jug A : 0 and Jug B : 2
 • Pour water in Jug A from Jug B    Then, Jug A : 2 and Jug B : 0
Goal Achieved

true.
```

(2,3)



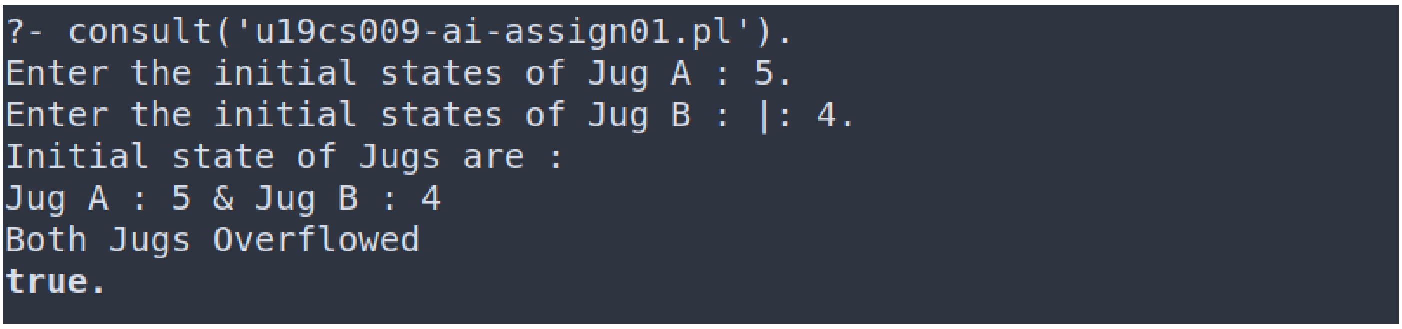
```
?- consult('u19cs009-ai-assign01.pl').
Enter the initial states of Jug A : 2.
Enter the initial states of Jug B : |: 3.
Initial state of Jugs are :
Jug A : 2 & Jug B : 3
 • Empty Jug B                        Then, Jug A : 2 and Jug B : 0
Goal Achieved

true.
```

(3,3)



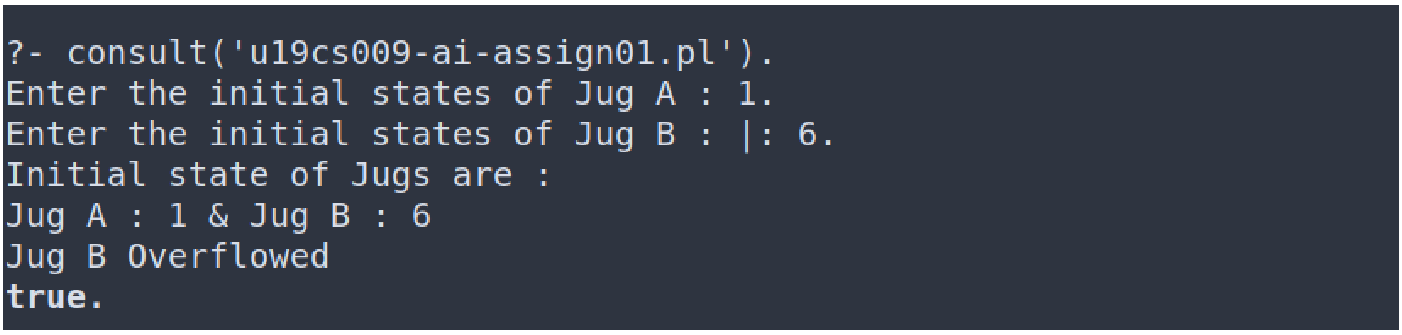
```
?- consult('u19cs009-ai-assign01.pl').
Enter the initial states of Jug A : 3.
Enter the initial states of Jug B : |: 3.
Initial state of Jugs are :
Jug A : 3 & Jug B : 3
 • Pour water in Jug A from Jug B     Then, Jug A : 4 and Jug B : 2
 • Empty Jug A                        Then, Jug A : 0 and Jug B : 2
 • Pour water in Jug A from Jug B     Then, Jug A : 2 and Jug B : 0
Goal Achieved

true.
```

Error (5,4)



```
?- consult('u19cs009-ai-assign01.pl').
Enter the initial states of Jug A : 5.
Enter the initial states of Jug B : |: 4.
Initial state of Jugs are :
Jug A : 5 & Jug B : 4
Both Jugs Overflowed
true.
```

Error (1,6)



```
?- consult('u19cs009-ai-assign01.pl').
Enter the initial states of Jug A : 1.
Enter the initial states of Jug B : |: 6.
Initial state of Jugs are :
Jug A : 1 & Jug B : 6
Jug B Overflowed
true.
```