

Department of Computer Science and Engineering  
Jahangirnagar University  
Savar, Dhaka



## **Laboratory Report**

CSE-404: Artificial Intelligence Laboratory

### **Submitted by**

Md. Nahid Hossain  
Exam Roll: 160060  
Class Roll: 54  
Session: 2015-16  
4th year 1st Semester

### **Submitted to**

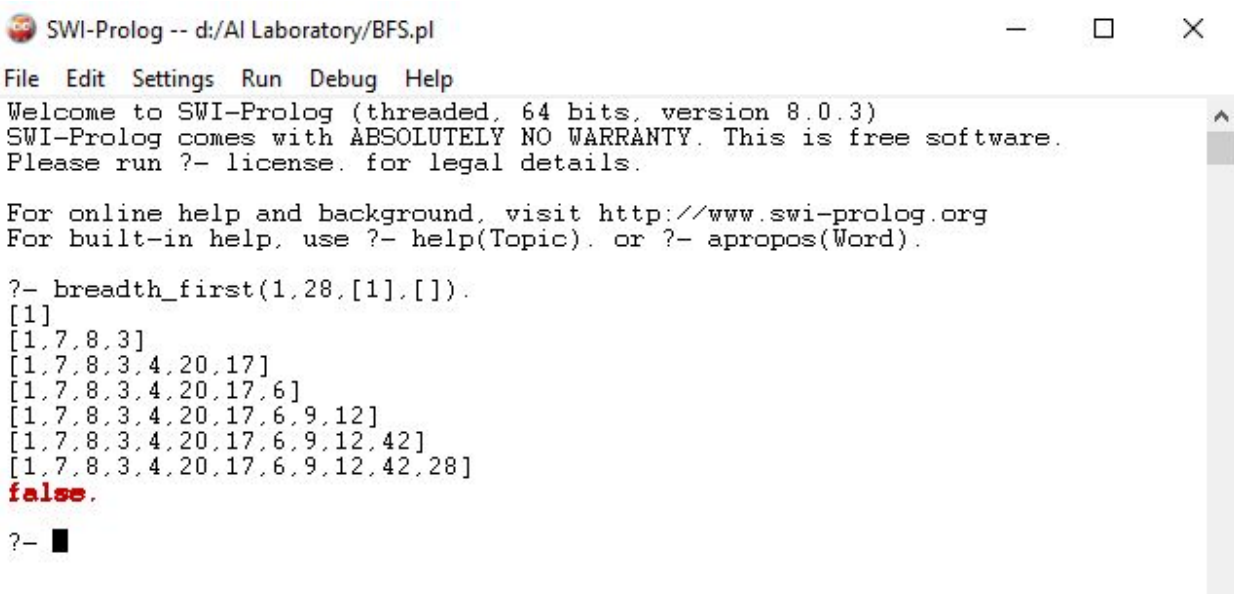
Dr. Sharif Uddin  
Professor  
Department of Computer Science and Engineering  
Jahangirnagar University

## Experiment Name: Implementation Of BFS and DFS in prolog.

### Prolog Code BFS:

```
1 connected(1,7,1).
2 connected(1,8,1).
3 connected(1,3,1).
4 connected(7,4,1).
5 connected(7,20,1).
6 connected(7,17,1).
7 connected(8,6,1).
8 connected(3,9,1).
9 connected(3,12,1).
10 connected(9,19,1).
11 connected(4,42,1).
12 connected(20,28,1).
13 connected(17,10,1).
14
15 connected2(X,Y,D) :- connected(X,Y,D).
16 connected2(X,Y,D) :- connected(Y,X,D).
17
18 next_node(Current, Next, Path) :-
19     connected2(Current, Next, _),
20     not(member(Next, Path)).
21
22 breadth_first(Goal, Goal, _, [Goal]).
23 breadth_first(Start, Goal, Visited, Path) :-
24     findall(X,
25         (connected2(X,Start,_, not(member(X,Visited)))),
26         [T|Extend]),
27     write(Visited), nl,
28     append(Visited, [T|Extend], Visited2),
29     append(Path, [T|Extend], [Next|Path2]),
30     breadth_first(Next, Goal, Visited2, Path2).
31
```

Figure 01: Prolog Code



```
SWI-Prolog -- d:/AI Laboratory/BFS.pl
File Edit Settings Run Debug Help
Welcome to SWI-Prolog (threaded, 64 bits, version 8.0.3)
SWI-Prolog comes with ABSOLUTELY NO WARRANTY. This is free software.
Please run ?- license. for legal details.

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

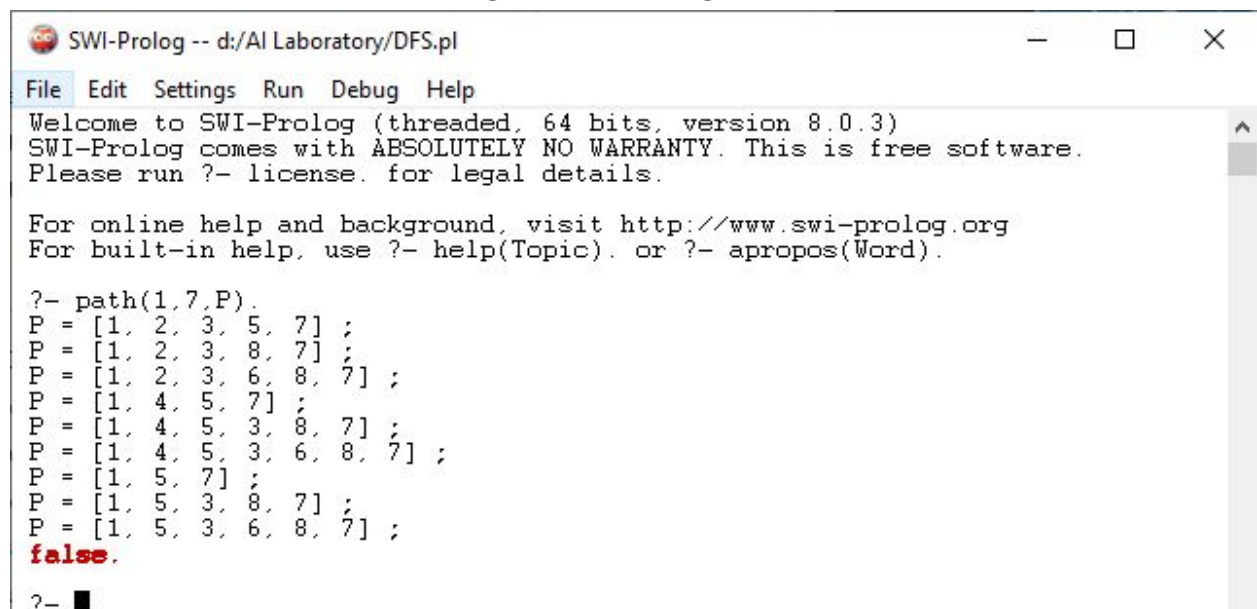
?- breadth_first(1,28,[1],[]).
[1]
[1,7,8,3]
[1,7,8,3,4,20,17]
[1,7,8,3,4,20,17,6]
[1,7,8,3,4,20,17,6,9,12]
[1,7,8,3,4,20,17,6,9,12,42]
[1,7,8,3,4,20,17,6,9,12,42,28]
false.
?- █
```

Figure 02: Result

### Prolog Code BFS:

```
1 edge(1, 2).
2 edge(1, 4).
3 edge(1, 5).
4 edge(2, 3).
5 edge(4, 5).
6 edge(3, 5).
7 edge(5, 7).
8 edge(3, 8).
9 edge(3, 6).
10 edge(8, 6).
11 edge(8, 7).
12 %edge(X,Y) :- edge(Y,X) , ! .
13 connected(X,Y) :- edge(X,Y) ; edge(Y,X) .
14
15 path(A,B,Path) :-
16     traverse(A,B,[A],Q),
17     reverse(Q,Path) .
18 traverse(A,B,P,[B|P]) :-
19     connected(A,B) .
20 traverse(A,B,Visited,Path) :-
21     connected(A,C),
22     C \== B,
23     \+member(C,Visited),
24     traverse(C,B,[C|Visited],Path) .
```

Figure 03: Prolog Code



```
SWI-Prolog -- d:/AI Laboratory/DFS.pl
File Edit Settings Run Debug Help
Welcome to SWI-Prolog (threaded, 64 bits, version 8.0.3)
SWI-Prolog comes with ABSOLUTELY NO WARRANTY. This is free software.
Please run ?- license. for legal details.

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

?- path(1,7,P).
P = [1, 2, 3, 5, 7] ;
P = [1, 2, 3, 8, 7] ;
P = [1, 2, 3, 6, 8, 7] ;
P = [1, 4, 5, 7] ;
P = [1, 4, 5, 3, 8, 7] ;
P = [1, 4, 5, 3, 6, 8, 7] ;
P = [1, 5, 7] ;
P = [1, 5, 3, 8, 7] ;
P = [1, 5, 3, 6, 8, 7] ;
false.
?-
```

Figure 04:Result