```
:- use_module(library(pce)).

% Sample graph
edge(a, b).
edge(a, c).
edge(b, d).
edge(b, e).
edge(c, f).
edge(e, g).

% DFS algorithm
dfs(Start, Goal, Path) :-
   dfs_helper(Start, Goal, [Start], Path).

dfs_helper(Goal, Goal, Path, Path).
dfs_helper(Node, Goal, Visited, Path) :-
   edge(Node, Next),
   \+ member(Next, Visited),
   dfs_helper(Next, Goal, [Next|Visited], Path).

% GUI to show DFS path
go :-
   new(Dialog, dialog('DFS GUI')),
   send(Dialog, append, new(StartLabel, text_item(start_node, 'a'))),
   send(Dialog, append, new(GoalLabel, text_item(goal_node, 'g'))),
   send(Dialog, append, button(run, message(@prolog, run_dfs, StartLabel?selection,
GoalLabel?selection))),
   send(Dialog, open).

run_dfs(Start, Goal) :-
   (dfs(Start, Goal, Path) ->
      reverse(Path, RevPath),
      format('DFS Path: ~w~n', [RevPath]),
      display_result(RevPath)
   ;
      send(@display, inform, 'No path found')
   ).

display_result(Path) :-
   (get(@display, exists) -> true ; new(@display, dialog('DFS Result'))),
   send(@display, clear),
   send(@display, append, text(Path)),
   send(@display, open).
```

File Edit Settings Run Debug Help

```
Welcome to SWI-Prolog (threaded, 64 bits, version 9.2.9)
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).

?- working_directory(_, 'D:/Prolog').
true.

?- [dfs_gui].
true.

?- go.
true.

?-
```