Q1.

footpath(52,53).

footpath(52,0).

footpath(53,54).

footpath(56,57).

footpath(56,58).

footpath(57,58).

footpath(57,1).

footpath(57,59).

footpath(58,59).

footpath(59,60).

footpath(83,65).

footpath(65,68).

footpath(76,69).

motorway(0,62).

motorway(53,62).

motorway(54,56).

motorway(54,62).

motorway(1,62).

motorway(59,60).

motorway(59,68).

motorway(59,83).

motorway(83,62).

motorway(68,66).

motorway(68,62).

motorway(68,63).

motorway(68,76).

motorway(68,64).

motorway(66,63).

motorway(66,69).

motorway(66,76).

motorway(66,64).

motorway(63,62).

motorway(63,64).

motorway(63,76).

motorway(63,69).

motorway(64,69).

motorway(64,76).

motorway(0,59).

motorway(0,60).

motorway(0,68).

motorway(0,83).

motorway(1,54).

routeFoot(A,B) :-

footpath(B,A);

footpath(A,B).

routeMotor(A,B) :-

motorway(A,B);

motorway(B,A).

routeMixed(A,Connector,B):-

footpath(A,Connector);

motorway(Connector,B);

footpath(Connector,A);

motorway(B,Connector).

footpathSearch(A,B,V) :-

(routeFoot(A,X),

not(member(X,V)),

(B=X;footpathSearch(X,B,[A|V]))).

motorwaySearch(A,B,V):-

(routeMotor(A,X),

not(member(X,V)),

(B=X;motorwaySearch(X,B,[A|V]))).

mixedSearch(A,B,C,V):-

(routeMixed(A,C,B),

not(member(X,V))),

(B=X;mixedSearch(A,B,C,[A|V])).

footpathRoute(A,B) :-

footpathSearch(A,B,[]).

motorwayRoute(A,B):-

motorwaySearch(A,B,[]).

mixedRoute(A,B,C):-

mixedSearch(A,B,C,[]).

route(A,B):-

footpathRoute(A,B);motorwayRoute(A,B);mixedRoute(A,B,0);mixedRoute(A,B,1).

highwayDesirable(A,B):-

\+(motorwayRoute(A,B);mixedRoute(A,B,0);mixedRoute(A,B,1)).

The above codes are for searching different kinds of paths from A to B. The function “footpathRoute(A,B) :-

footpathSearch(A,B,[]).

” is used to find paths consist of footpath only, and the function “motorwayRoute(A,B):-

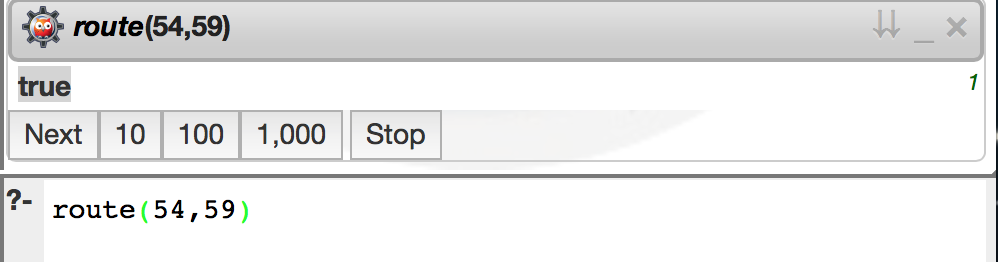
motorwaySearch(A,B,[]).

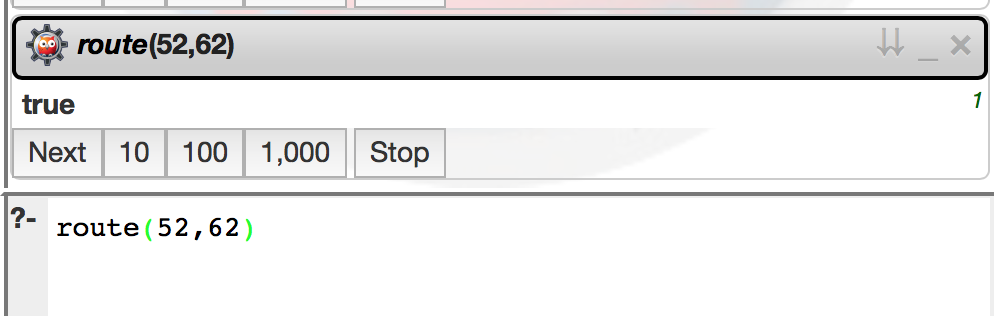
” is used to find paths consist of motorway only, and the function “mixedRoute(A,B,C):-

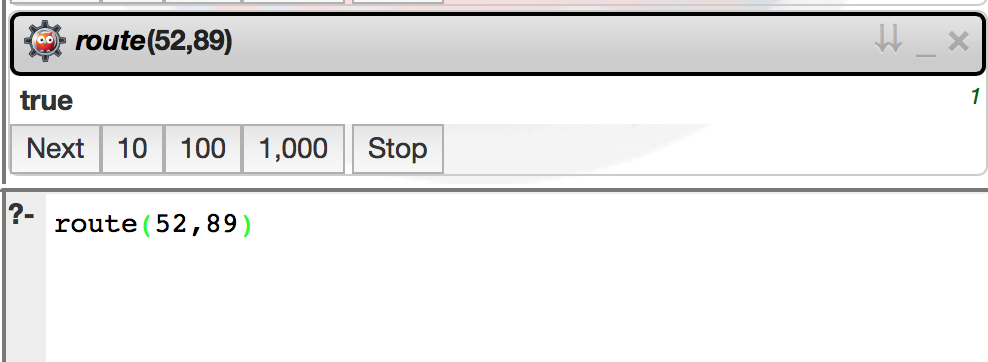
mixedSearch(A,B,C,[]).

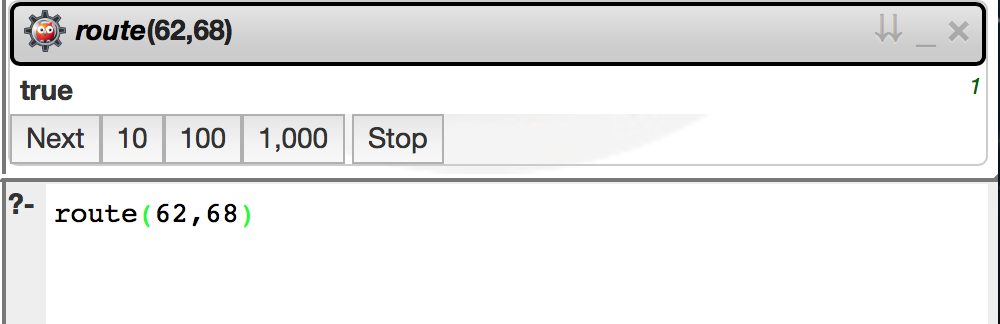
” is used to find paths consist of footpath and footpath. Here C refers to the connector(the link point of single line and double line), and C=0 refers to the connector connected to building 52 while C=1 refers to the connector connected to building 57

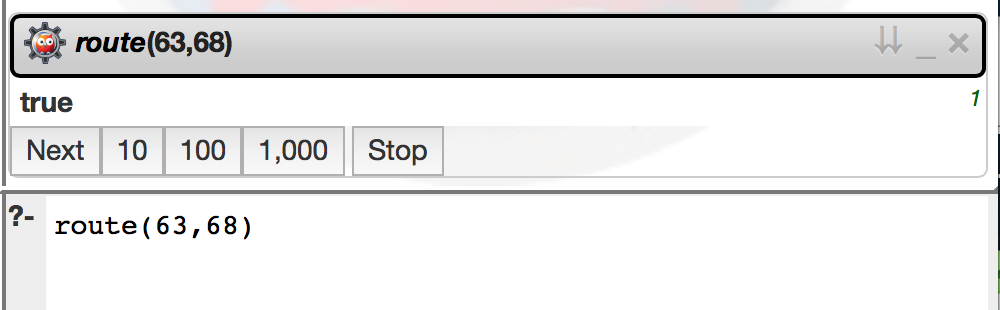
b)









:

c)

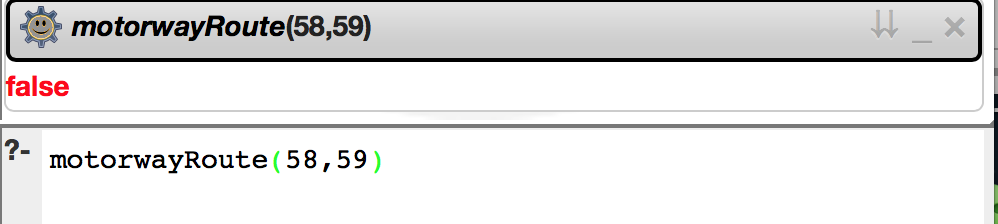
* call the method

motorwayRoute(A,B):-

motorwaySearch(A,B,[]).

and set A = 58, B=59

it will return whether there is an motorway connecting A and B



* call the method

highwayDesirable(A,B):-

\+(motorwayRoute(A,B);mixedRoute(A,B,0);mixedRoute(A,B,1)).

it will return whether a highway required. Here as long as one segment of the path from A to B is high way, it will not require a highway.

Q2：

wife(elizabeth,philip).

wife(diana,charles).

wife(camilla,charles).

wife(anne,mark).

wife(anne,timothy).

wife(sarah,andrew).

wife(sophie,edward).

wife(kate,william).

wife(autumn,peter).

wife(zara,mike).

son(charles, philip).

son(charles, elizabeth).

daughter(anne, philip).

daughter(anne, elizabeth).

son(andrew, philip).

son(andrew, elizabeth).

son(edward, philip).

son(edward, elizabeth).

son(william, charles).

son(william, diana).

son(harry, charles).

son(harry, diana).

son(peter, mark).

son(peter, anne).

daughter(zara, mark).

daughter(zara, anne).

daughter(beatrice, andrew).

daughter(beatrice, sarah).

daughter(eugenie, andrew).

daughter(eugenie, sarah).

daughter(louise, edward).

daughter(louise, sophie).

son(james, edward).

son(james, sophie).

son(george, william).

son(george, kate).

daughter(savannah, peter).

daughter(savannah, autumn).

daughter(isla, peter).

daughter(isla, autumn).

daughter(mia, mike).

daughter(mia, zara).

mother(X, C):- child(C, X),wife(X, P).

husband(Husband,Wife):- wife(Wife,Husband).

spouse(A,B):- wife(A,B); husband(A,B).

child(Child,Parent):- son(Child,Parent); daughter(Child,Parent).

parent(Parent,Child):- child(Child,Parent).