1.1 Does the graph in Figure 1 have any spanning spider? If yes, indicate the center vertex and the list of edges included in the spanning spider.

Yes, it does.

center(z)

edge(z,y), edge(y,b)

edge(z,x), edge(x,e)

edge(z,c), edge(c,d), edge(d,a)

1.2 Provide ASP rules that define the center/1 predicate and ensure that in any model, exactly one vertex is selected as the center.

If there is a vertex with degree 3 in the graph, then we should choose that vertex as center,

else we choose any vertex as center.

$$edges\ (X,Y) :- edge(X,Y).$$
$$edges\ (X,Y) :- edges(Y,X).$$
$$center1(A) :- edges(X,A), edges(Y,A), edges(Z,A), X\ !=\ Y, Y\ !=\ Z, X\ !=\ Z.$$
$$1\ \{center(X) : center1(X)\}\ 1 :- center1(Y).$$
$$1\ \{center(X) : vertex(X)\} 1 :- not\ center1(Y), vertex(Y).$$

1.3 Provide a generator for the leg/2 predicate.

The leg is the selected edge.

$$\{leg(X,Y)\} :- edge(X,Y).$$

1.4 One property required of spanning spiders is that every vertex should be reachable from the center through leg edges. Introduce a derived predicate reachable/1 that ranges over vertex names and is true when the corresponding vertex is reachable from the center through leg edges. Use this predicate to de ne a constraint that ensures every vertex of the original graph is reachable from the center through leg edges.

We need to make sure that none of the vertices are isolated.

$$path(X,Y) :- edges(X,Y).$$
$$path(X,Z) :- path(X,Y), path(Y,Z).$$
$$reachable(Y) :- path(X,Y), center(X), vertex(Y).$$

1.5 The reachability of every vertex from the center is a required property of spanning spiders, but it is not sufficient and we need to ensure other constraints are satisfied. Describe all the other constraints that need to be satisfied and write ASP rules to enforce these constraints.

The center must have the number of legs above or equal to 2.

The other vertices must have the number of legs below or equal to 2.

Every vertex should have connected leg.

There should not be a leg isolated.

The graph should be reachable.

There should not be a circle leg.

$$degree\ (V,D) :- vertex(V), D\ =\ \#count\{\ U : leg(U,V);\ U : leg(V,U)\ \}.$$
$$:- center(X), degree(X,D), D < 2.$$
$$:- not\ center(X), degree(X,D), D > 2.$$

$$:- degree\ (X, D), vertex(X), D\ =\ 0.$$
$$legs(X, Y) :- leg(X, Y).$$
$$legs(X, Y) :- legs(Y, X).$$
$$legs(X, Z) :- legs(X, Y), legs(Y, Z).$$
$$:- not\ legs(X, Y), vertex(X), vertex(Y).$$
$$:- not\ reachable(X), vertex(X).$$
$$numv(X) :- X\ =\ \#count\{\ V: vertex(V)\ \}.$$
$$numl(X) :- X\ =\ \#count\{\ V, W: leg(V, W)\ \}.$$
$$:- numv(X), numl(Y), not\ Y\ <\ X.$$

1.6 Based on your answers to the above questions, write an ASP program spiders.lp that takes an input graph and outputs all the distinct spanning spiders of the graph. How many distinct spanning spiders does the graph in Figure 1 have?

There are 54 distinct spanning spiders in Figure 1.

1.7 Spanning spiders with short legs. Write an ASP program spidershortlegs.lp that outputs a spanning spider with the shortest longest leg.