



在第六周结束时到期。10月21日，星期五，晚上11:55。这是一项个人作业。
每个学生提交：assn1.pdf、spiders.lp、spidershortlegs.lp。
assn1.pdf包含你对问题的回答，包括任何相关的代码位。
spiders.lp和spidershortlegs.lp包含了你完成的全部程序（它不应该包括样本数据，如通过顶点/1和边缘/2声明一个特定的图形，但它应该包括打印出所需输出的一行）。

这项作业满分为10分，占课程成绩的10%。

问题1 跨越式蜘蛛

一个图的生成树是一个子图，它是一棵树，包含原图的所有顶点。蜘蛛图是一个最多只有一个顶点的度数在3以上的图，这个顶点被称为蜘蛛图的 *中心*。在这个练习中，我们感兴趣的是创建一个程序来寻找输入图形的 *生成蛛网*：对于任何图形，我们要找出一个既是生成树又是蛛网的子图。在一个生成蛛网中，我们把包括子图在内的边称为 *腿边*。

这个练习的灵感来自于2003年Prolog编程竞赛中的一个问题。¹我们的目标是编写一个答案集程序，使得给定一个图作为输入，每个稳定的模型都对应于一个不同的跨越式蜘蛛。并非每个图都有一个跨越式蜘蛛，所以如果一个输入的图不承认任何生成蛛，相应的程序应该是不可满足的。

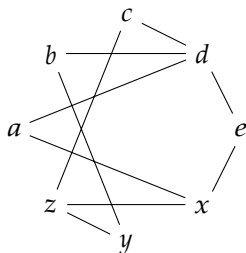


图1：示例图

输入格式 一个输入文件包含表示图形顶点的vertex/1的谓词实例和表示图形边的edge/2的谓词实例。

输出格式 你的程序应该计算出一个模型，其中包含一个谓词leg/2，表示哪些边被包含在跨越式蜘蛛中，以及一个谓词center/1，表示哪个顶点是蜘蛛的中心。

```

顶点 ( a
) 。 顶点
( b ) 。
顶点 ( c
) 。 顶点
( d ) 。
顶点 ( e
) 。 顶点
( x ) 。
顶点 ( y
) 。 顶点
( z ) 。
边缘 ( a ,
d ) 。
边缘(a, x) 。
边缘(b, d) 。
边缘(b, y) 。
边缘(c, d) 。

```

¹ <https://people.cs.kuleuven.be/~bart.demoen/PrologProgrammingContests/Contest2003.html>

边缘(c, z) ◦
边缘(d, e) ◦
边缘(e, x) ◦
边缘(x, z) ◦
边缘(y, z) ◦

1.1 图1中的图形是否有任何生成蛛网？如果有，请指出中心顶点和生成蛛网中包含的边的列表。

1.2 提供定义中心/1谓词的ASP规则，并确保在任何模型中，正好有一个顶点被选为中心。

1.3 为leg/2谓词提供一个生成器。

1.4 跨越式蜘蛛的一个特性是，每个顶点都可以通过腿边从中心到达。引入一个派生谓词 `reachable/1`，它的范围是顶点名称，当相应的顶点通过腿边可以从中心到达时为真。使用这个谓词来定义一个约束，确保原图的每个顶点都可以通过腿边从中心到达。

1.5 从中心到每个顶点的可及性是跨越式蜘蛛的必要属性，但这还不够，我们需要确保其他约束条件得到满足。描述所有需要满足的其他约束条件，并编写ASP规则来执行这些约束。

1.6 根据你对上述问题的回答，编写一个ASP程序 `spiders.lp`，该程序接受一个输入图，并输出该图的所有不同跨度的 `spiders`。图1中的图有多少个不同的跨期蜘蛛？

1.7 有短腿的跨越式蜘蛛。编写一个ASP程序 `spidershortlegs.lp`，输出一个具有最短长腿的跨越式蜘蛛。