**习题4.1**

**代码：**

clc,clear,close all

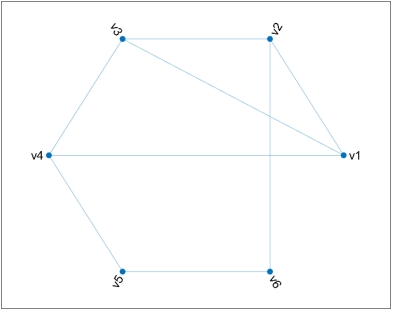
a1=zeros(6);

a1(1,[2:4])=1;a1(2,[3,6])=1;a1(3,4)=1;a1(4,5)=1;a1(5,6)=1;

s=cellstr(strcat('v',int2str([1:6]')));

G1=graph(a1,s,'upper');

plot(G1,'Layout','circle')



a2=zeros(6);

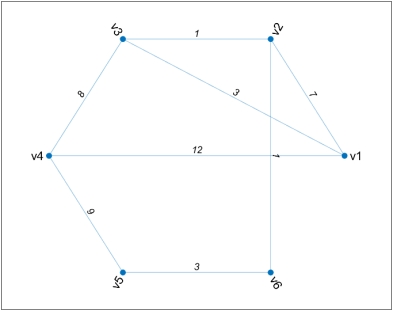
a2(1,[2:4])=[7,3,12];a2(2,[3,6])=[1,1];

a2(3,4)=8;a2(4,5)=9;a2(5,6)=3;

s=cellstr(strcat('v',int2str([1:6]')));

G2=graph(a2,s,'upper');

plot(G2,'Layout','circle','EdgeLabel',G2.Edges.Weight)



a3=zeros(6);

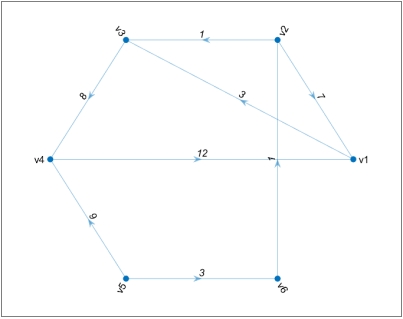
a3(1,3)=3;a3(2,[1,3])=[7,1];a3(3,4)=8;

a3(4,1)=12;a3(5,[4,6])=[9,3];a3(6,2)=1;

s=cellstr(strcat('v',int2str([1:6]')));

G3=digraph(a3,s);

plot(G3,'EdgeLabel',G3.Edges.Weight,'Layout','circle')



**习题4.4**

**问题分析：**

这是一个最短路径的问题，可以使用Dijkstra标号算法求解。

**符号说明：**

分别用p,d表示最短路径和最短距离。

**模型建立：**

1. 首先从v1出发，v1到v1的最短距离为0，标记节点1
2. 从v1出发，到v2的距离为20，到v5的距离为15，节点2和节点5更新，其前面点均为节点1，标记节点5
3. 从v1出发，经过v5,到v2的距离为40，到v3的距离为33，到v4的距离为50，到v6的距离为30，节点3、节点4和节点6更新，其前面点均为节点5，标记节点2
4. 从v1出发，经过v2,到v3的距离为40，到v4的距离为80，到v5的距离为45，不更新任何节点，标记节点6
5. 从v1出发，经过v6,到v4的距离为40，节点4更新，其前面点为节点6，标记节点3
6. 从v1出发，经过v3,到v4的距离为63，不更新任何节点，标记节点4，结束。

求得从v1到v4的最短路径为v1→v5→v6→v4，最短距离为40。

**代码：**

clc,clear,close all

a=zeros(6);

a(1,[2,5])=[20,15];a(2,[3:5])=[20,60,25];

a(3,[4,5])=[30,18];a(4,[5,6])=[35,10];a(5,6)=15;

s=cellstr(strcat('v',int2str([1:6]')));

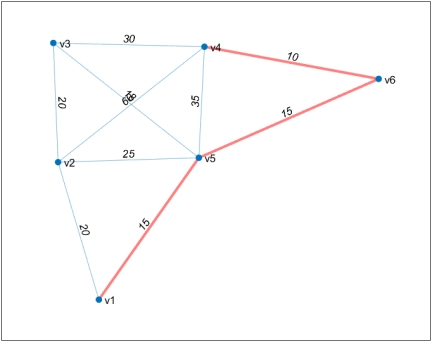
G=graph(a,s,'upper');

[p,d]=shortestpath(G,1,4)

h=plot(G,'EdgeLabel',G.Edges.Weight);

highlight(h,p,'EdgeColor','r','LineWidth',2)

disp('(d)赋权无向图');



**运行结果：**

p = 1×4

1 5 6 4

d = 40