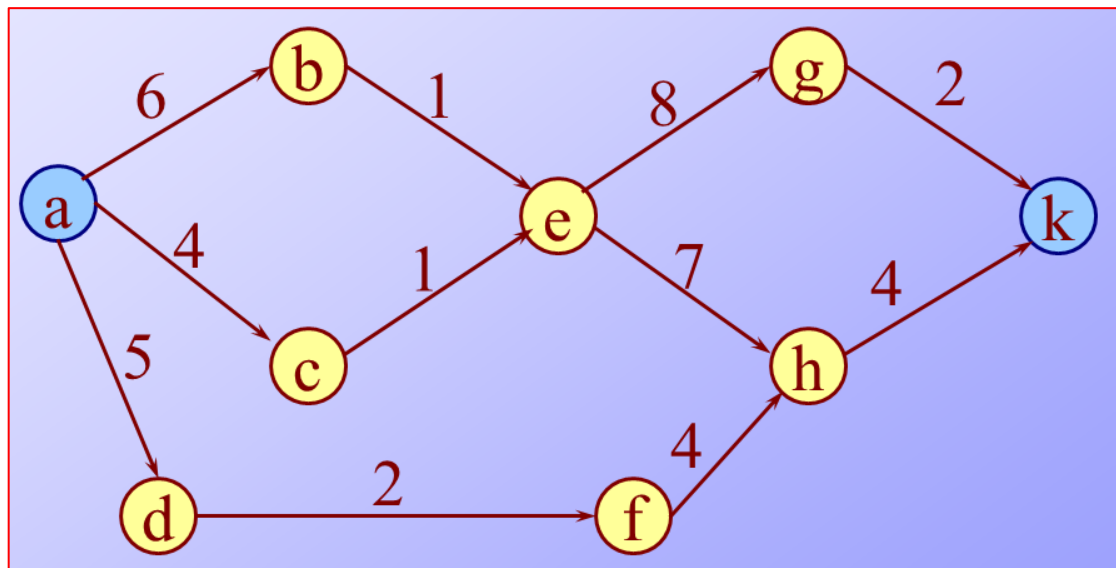


一、输入为有向网时候，输出为邻接表和逆邻接表,拓扑排序及关键路径



输入:

请输入图的类型(0-有向图, 1-有向网, 2-无向图, 3-无向网)

1

G.vexnum :9

G.arcnum :11

G.vertices[0].data : a

G.vertices[1].data : b

G.vertices[2].data : c

G.vertices[3].data : d

G.vertices[4].data : e

G.vertices[5].data : f

G.vertices[6].data : g

G.vertices[7].data : h

G.vertices[8].data : k

v1 (char) : a

v2 (char) : b

w (int) : 6

v1 (char) : a

v2 (char) : c

w (int) : 4

v1 (char) : a

v2 (char) : d

w (int) : 5

v1 (char) : b

v2 (char) : e

w (int) : 1

v1 (char) : c

v2 (char) : e

w (int) : 1

v1 (char) : d

v2 (char) : f

w (int) : 2

v1 (char) : e

v2 (char) : g

w (int) : 8

v1 (char) : e

v2 (char) : h

w (int) : 7

v1 (char) : f

v2 (char) : h

w (int) : 4

v1 (char) : g

v2 (char) : k

w (int) : 2
v1 (char) : h
v2 (char) : k
w (int) : 4

输出:

邻接表为 G

有向网

0|a|->|3,d,5|->|2,c,4|->|1,b,6|->NULL
1|b|->|4,e,1|->NULL
2|c|->|4,e,1|->NULL
3|d|->|5,f,2|->NULL
4|e|->|7,h,7|->|6,g,8|->NULL
5|f|->|7,h,4|->NULL
6|g|->|8,k,2|->NULL
7|h|->|8,k,4|->NULL
8|k|->NULL

逆邻接表为 InverseG

有向网

0|a|->NULL
1|b|->|0,a,6|->NULL
2|c|->|0,a,4|->NULL
3|d|->|0,a,5|->NULL
4|e|->|2,c,1|->|1,b,1|->NULL
5|f|->|3,d,2|->NULL
6|g|->|4,e,8|->NULL
7|h|->|5,f,4|->|4,e,7|->NULL
8|k|->|7,h,4|->|6,g,2|->NULL

拓扑排序的结果为

0(a) 1(b) 2(c) 4(e) 6(g) 3(d) 5(f) 7(h) 8(k)

有向网的关键路径为:

0(a)---1(b)(dut=6,ee=0,el=0,tag=*)
1(b)---4(e)(dut=1,ee=6,el=6,tag=*)
4(e)---7(h)(dut=7,ee=7,el=7,tag=*)
7(h)---8(k)(dut=4,ee=14,el=14,tag=*)

请按任意键继续...

注意: 输入为有向图时候, 输出为邻接表和逆邻接表及拓扑排序, 这里就不举例了。通过上个例子就可以了解全部了。