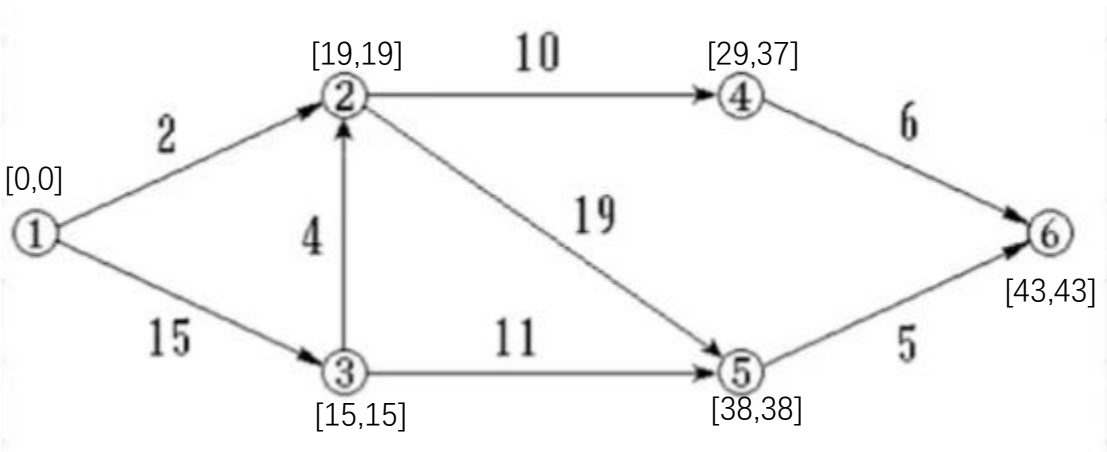


1、 拓扑排序与关键路径的重要例题：

请对图 1 所示的 AOE 网进行如下计算：

- 1) 求拓扑排序序列与逆拓扑排序序列；
- 2) 求这个工程最早可能在什么时间结束；
- 3) 求每个活动的最早开始时间和最迟开始时间；
- 4) 确定哪些活动是关键活动。



解答：

- 1) 求拓扑排序时，只需每次找出入度为 0 的结点，并输出该结点，最后从图中删除该结点以及以该结点为弧尾的所有弧，如此迭代，直至图中不存在入度为 0 的结点。若最后输出结点的数目小于有向图中的顶点数，则说明有向图中存在回路，否则输出的顶点的顺序即为一个拓扑排序。拓扑排序序列并不唯一，但是具有先决关系的结点，其相对顺序会出现在所有的拓扑排序序列之中。

Topological sorting 1: 1 -> 3 -> 2 -> 4 -> 5 -> 6

Topological sorting 2: 1 -> 3 -> 2 -> 5 -> 4 -> 6

-----

求逆拓扑排序时，只需每次找出出度为 0 的结点，并输出该结点，最后从图中删除该结点以及以该结点为弧头的所有弧，如此迭代，直至图中不存在出度为 0 的结点。

Reverse Topological sorting 1: 6 -> 4 -> 5 -> 2 -> 3 -> 1

Reverse Topological sorting 2: 6 -> 5 -> 4 -> 2 -> 3 -> 1

- 2) 下面开始推算各个事件的最早发生时间（具体做法：从前往后推，前驱结点（事件）到当前结点（事件）所需时间，取最大值）。

事件 1（结点 1）的最早发生时间为 0；

事件 3（结点 3）最早的发生时间为 15；

事件 2（结点 2）最早的发生时间为 19，因为事件 2（结点 2）发生需要满足两个条件，活动 <v1,v2>和活动<v1,v3>+<v3,v2>都完成，则由耗时长的活动决定，则事件 2（结点 2）的最早发生时间是由活动<v1,v3>+<v3,v2>决定，为 15+4=19；

事件 5（结点 5）最早的发生时间为 38，因为事件 5（结点 5）发生需要满足两个条件，活动 <v2,v5> 和活动 <v3,v5> 都完成。活动 <v2,v5>的最早发生时间 = 事件 2（结点 2）最早发生时间+ 活动<v2,v5> 耗时 = 19+19 = 38 。活动 <v3,v5>的最早发生时间 = 事件 3（结点 3）最早发生时间+ 活动<v3,v5> 耗时 = 15+11 = 26 。耗时长的活动决定事件 5（结点 5）最早的发生时间，则为 38 。

事件 4（结点 4）最早的发生时间为 29，因为事件 4（结点 4）发生需要满足一个条件，即活动 <v2,v4> 完成。活动 <v2,v4>的最早发生时间 = 事件 2（结点 2）最早发生时间+ 活动<v2,v4> 耗时 = 19+10 = 29 。

事件 6（结点 6）最早的发生时间为 43，因为事件 6（结点 6）发生需要满足两个条件，活动 <v4,v6> 和活动 <v5,v6> 都完成。活动 <v4,v6>的最早发生时间 = 事件 4（结点 4）最早发生时间+ 活动<v4,v6> 耗时 = 29+6 = 35 。活动 <v5,v6>的最早发生时间 = 事件 5（结点 5）最早发生时间+ 活动<v5,v6> 耗时 = 38+5 = 43 。耗时长的活动决定事件 6（结点 6）最早的发生时间，则为 43 。

因此，这个工程最早结束时间为 43。

3) 上一问，我们已经将各个事件的最早发生时间求出。现在将其依次填入表格 1 的第二行（Ve），另外，为方便后续分析，我们在事件（顶点）旁边加了一个 '[x,y]'，其中 x 为这个事件的最早发生时间，y 为这个事件的最晚发生时间。因此，相应更新 [x,y] 中的 x 值。

		V1	V2	V3	V4	V5	V6
事件最早发生时间	Ve	0	19	15	29	38	43
事件最晚发生时间	VI						

下面开始推算各个事件的最晚发生时间（具体做法：从后往前推，后继结点（事件）的最迟发生时间-边权重，取最小值）。

事件 6（结点 6）的最早、最晚发生时间都为 43；

事件 5（结点 5）的最晚发生时间为：事件 6（结点 6）的最晚发生时间 - 活动<v5,v6> 耗时 = 43-5 = 38；

事件 4（结点 4）的最晚发生时间为：事件 6（结点 6）的最晚发生时间 - 活动<v4,v6> 耗时 = 43 – 6 = 37；

事件 2（结点 2）的最晚发生时间为：min { 事件 4（结点 4）的最晚发生时间 - 活动<v2,v4>

耗时, 事件 5（结点 5）的最晚发生时间 - 活动<v2,v5> 耗时 } = min { 37 – 10, 38 – 19} = min {27, 19} =19；

事件 3（结点 3）的最晚发生时间为：

min {事件 5（结点 5）的最晚发生时间 –活动<v3,v5> 耗时, 事件 2（结点 2）的最晚发生时间 – 活动<v3,v2>耗时} = min { 38-11, 19-4} = min{27, 15} = 15；

事件 1（结点 1）的最晚发生时间为：

min {事件 2（结点 2）的最晚发生时间 –活动<v1,v2> 耗时, 事件 3（结点 3）的最晚发生时间 – 活动<v1,v3>耗时} = min { 19-2, 15-15} = min{17, 0} = 0；

此时我们已经将各个结点（事件）的最晚发生时间推算出来了。下面将其依次填入表 1 中的第二行，如下：

		V1	V2	V3	V4	V5	V6
事件最早发生时间	Ve	0	19	15	29	38	43
事件最晚发生时间	VI	0	19	15	37	38	43

4) 求各个活动（边）的最早开始时间和最晚开始时间。

活动<v4,v6>的最晚开始时间为 43-6 = 37。

活动<v5,v6>的最晚开始时间为 43-5 = 38。

-----

活动<v2,v4> 的最早开始时间为 19，因为<v2,v4>开始的条件是<v1,v2> 和 <v1,v3>+<v3,v2> 都结束，由耗时最长的活动决定，所以为<v1,v3>+<v3,v2> = 15+4 =19。

因为活动<v4,v6>的最晚开始时间为 43-6 = 37，活动<v,2,v4>是活动<v4,v6>的前驱，因此活动<v,2,v4>的最晚开始时间为 37-10 =27。

-----

活动<v4,v6>的最早开始时间为 29。因为活动<v2,v4> 最早开始时间为 19，则活动<v4,v6> 最早的开始时间为 19+10 = 29。

-----

活动<v2,v5> 的最早开始时间是 15+4 = 19， 因为<v2,v5>开始的条件也是<v1,v2> 和 <v1,v3>+<v3,v2> 都结束，由耗时最长的活动决定，所以为<v1,v3>+<v3,v2> = 15+4 =19。

活动<v2,v5> 的最晚开始时间是事件<v5,v6>最晚开始时间-19 = 38-19 = 19。

-----

活动<v3,v5> 的最早开始是 15，因为<v3,v5>唯一的前驱活动是<v1,v3>，其最早开始时间为 0，耗时为 15，所以活动<v3,v5> 的最早开始是 15。活动<v3,v5> 的最晚开始时间是活动

<v5,v6>的最晚开始时间 -<v3,v5>耗时 = 38 - 11 = 27 。

活动<v5,v6> 的最早开始时间是由 max {<v2,5>最早开始时间+19, <v3,v5>最早开始时间+11} 决定，则为 max{ 38, 26} = 38。

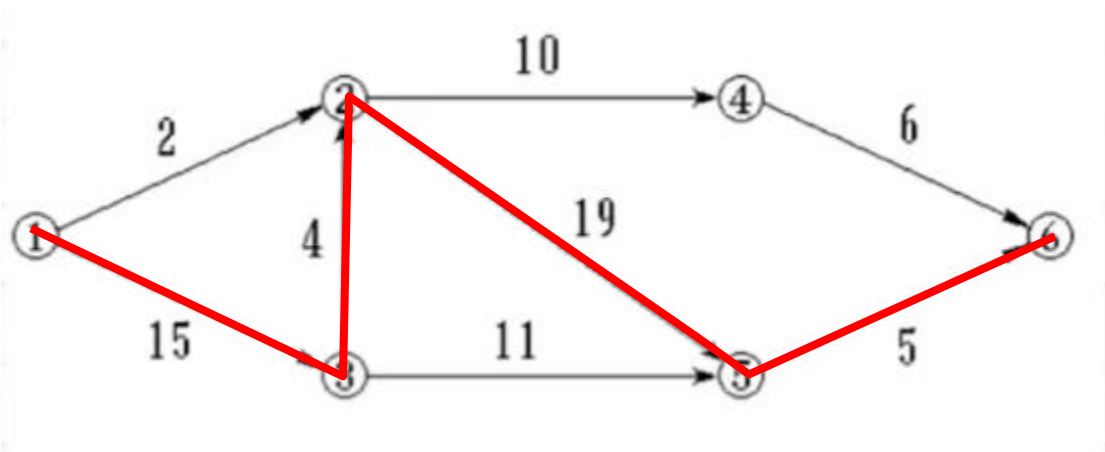
活动<v3,v2> 的最早开始是 15，因为<v3,v2>唯一的前驱活动是<v1,v3>，其最早开始时间为 0，耗时为 15，所以活动<v3,v2> 的最早开始是 15。活动<v3,v2> 的最晚开始时间是 Min{ 事件<V2,V5>最晚开始时间-4, 事件<V2,V4>最晚开始时间 -4 } = Min {19-4, 27-4} = Min{15, 23} = 15。

活动<v1,v2> 的最早开始是 0，活动<v1,v2> 的最晚始时间是 17，因为活动<v1,v2>和活动 <v1,v3>+<v3,v2>都是指向事件 2，而<v1,v3>+<v3,v2>耗时 15+4 =19，所以活动<v1,v2> 可以最晚在<v1,v3>+<v3,v2>运行到 17 时才开始。

活动<v1,v3> 的最早开始是 0，活动<v1,v3> 的最晚始时间是 0。

		<V1,V2>	<V1,V3>	<V2,V4>	<V2,V5>	<V3,V2>	<V3,V5>	<V4,V6>	<V5,V6>
活 动 最 早 发 生 时 间	e	0	0	19	19	15	15	29	38
活 动 最 晚 发 生 时 间	l	17	0	27	19	15	27	37	38
差 值	l-e	17	0	8	0	0	12	8	0

差值为 0 的活动为关键活动。分别为： <V1,V3>， <V2,V5>， <V3,V2>， <V5,V6>



如图红色标识的是这个工程的关键路径。