4. 网络流问题

定义1 设$G=(V, E)$为有向图，在V中指定一点称为**发点**(记为v_s)，和另一点称为**收点**(记为v_t)，其余点叫做中间点. 对每一条边$v_iv_j \in E$，对应一个非负实数C_{ij}，称为它的**容量**. 这样的G称为**容量网络**，简称**网络**，记作$G=(V, E, C)$.

定义2 网络$G=(V, E, C)$中任一条边v_iv_j有流量f_{ij}，称集合$f=\{f_{ij}\}$为网络G上的一个**流**.

满足下述条件的流f称为**可行流**：

① (限制条件) 对每一边v_iv_j，有$0 \leqslant f_{ij} \leqslant C_{ij}$；

② (平衡条件) 对于中间点v_k有$\sum f_{ik} = \sum f_{kj}$，即中间点$v_k$的输入量 = 输出量.

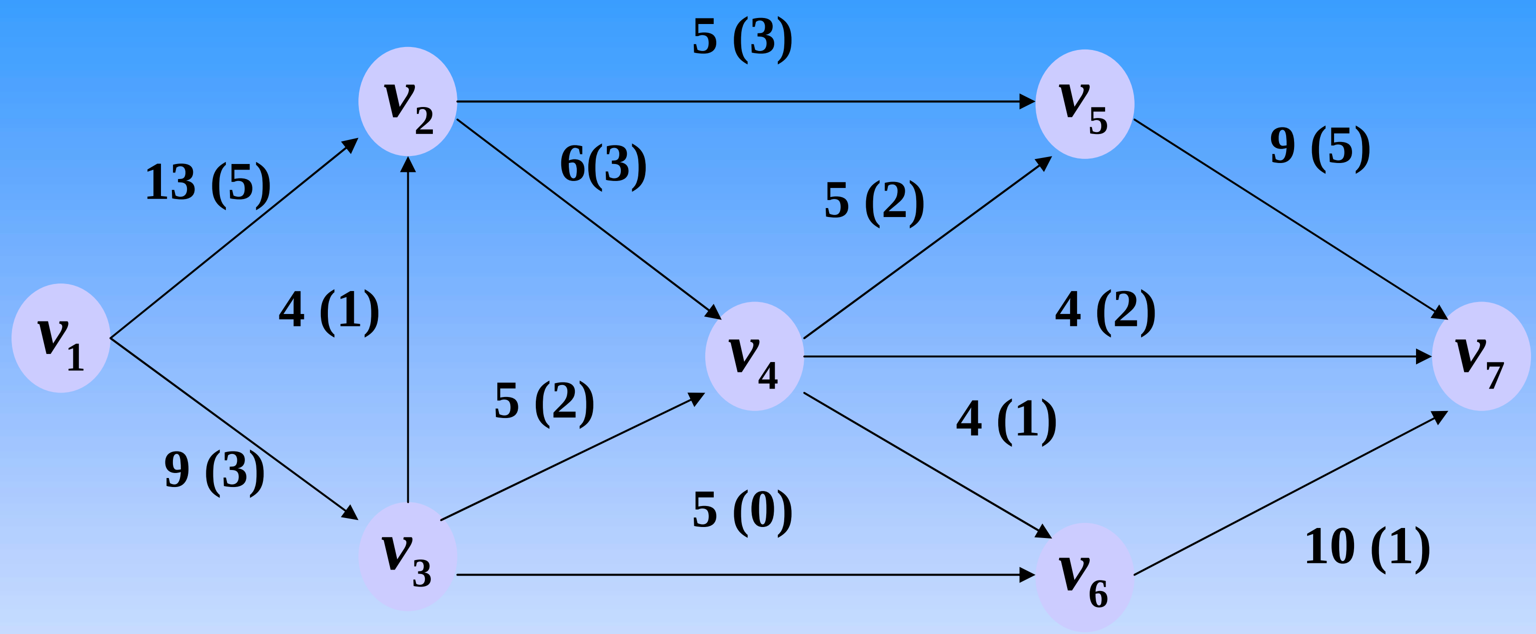
如果 f 是可行流, 则对收、发点v_t、v_s有

$$\sum f_{si}=\sum f_{jt}=W_f,$$

即从v_s点发出的物质总量 = v_t点输入的量. W_f称为网络流 f 的总流量.

上述概念可以这样来理解, 如G是一个运输网络, 则发点v_s表示发送站, 收点v_t表示接收站, 中间点v_k表示中间转运站, 可行流 f_{ij} 表示某条运输线上通过的运输量, 容量C_{ij}表示某条运输线能承担的最大运输量, W_f 表示运输总量.

可行流总是存在的. 比如所有边的流量 $f_{ij}=0$ 就是一个可行流(称为零流).



图中（v_3 ,v_6）为零流弧，其余为非饱和弧。

所谓最大流问题就是在容量网络中，寻找流量最大的可行流．求最大可行流的算法

实际问题中，一个网络会出现下面两种情况：(1) 发点和收点都不止一个．

解决的方法是再虚设一个发点v_s和一个收点v_t，发点v_s到所有原发点边的容量都设为无穷大，所有原收点到收点v_t边的容量都设为无穷大．

(2) 网络中除了边有容量外，点也有容量．

解决的方法是将所有有容量的点分成两个点，如点v有容量C_v，将点v分成两个点v'和v''，令

$$C(v'v'') = C_v .$$

最小费用流问题

这里我们要进一步探讨不仅要使网上的流达到最大，或者达到要求的预定值，而且还要使运输流的费用是最小的，这就是最小费用流问题.

最小费用流问题的一般提法：

已知网络$\boldsymbol{G = (V, E, C)}$，每条边$\boldsymbol{v_i v_j \in E}$除了已给容量$\boldsymbol{C_{ij}}$外，还给出了单位流量的费用$\boldsymbol{b_{ij}}$（$\geq$0）.

所谓最小费用流问题就是求一个总流量已知的可行流$\boldsymbol{f = \{f_{ij}\}}$使得总费用

$$b(f) = \sum_{v_i v_j \in E} b_{ij} f_{ij}$$

最小.

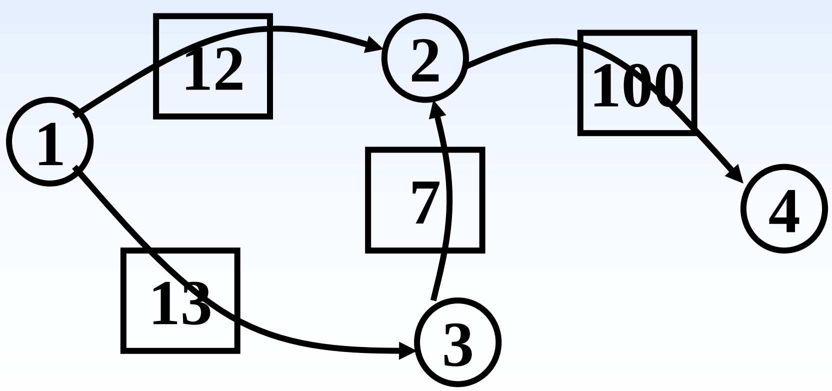
.特别地,当要求 f 为最大流时,此问题即为最小费用最大流问题

5 关键路径问题

一项工程任务, 大到建造一座大坝, 一座体育中心, 小至组装一台机床, 一架电视机, 都要包括许多工序. 这些工序相互约束, 只有在某些工序完成之后, 一个工序才能开始. 即它们之间存在完成的先后次序关系, 一般认为这些关系是预知的, 而且也能够预计完成每个工序所需要的时间.

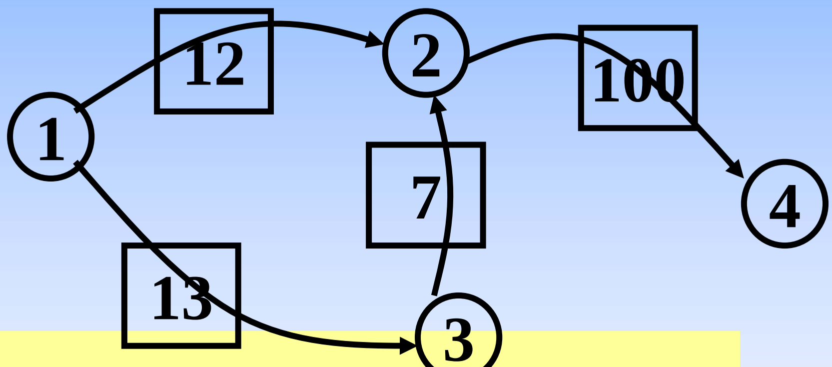
这时工程领导人员迫切希望了解最少需要多少时间才能够完成整个工程项目, 影响工程进度的要害工序是哪几个?

- 如下图，某个项目由4个作业（边）完成，每项作业需要一定时间（边的权值）完成，并且每项作业都需要在一定的状态（顶点）下才能开始，即要完成所有先行作业（所有进入该顶点的边）。求完成这个项目的最短时间。
- 无回路有向赋权图中的最长路径：关键路径。



- 关键路径问题图论中已有成熟的算法，具体可见数据结构或者图论方面的参考书。
- 数学规划方法，用Lingo解决：

设**xi**是作业**i**的开始时间。



目标：最后一个作业的开始时间最小。

$$\min z = x_n$$

$$st: x_j \geq x_i + t_{ij}, \quad t_{i,j} \neq 0, \quad i, j = 1, 2, \cdots, n$$

$$x_1 = 0, x_i \geq 0, \quad i = 1, 2, \cdots, n$$

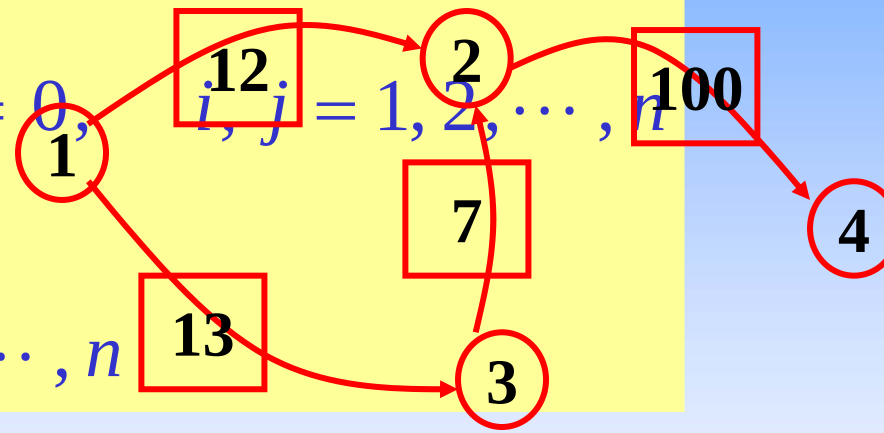
- 为了得到每个作业的最早开工时间和最迟开工时间，可更改模型如下：

$$\min z = \sum_{i=1}^{n} x_i$$ 全部作业的开始时间最小

$$st: s_{ij} = x_j - x_i - t_{ij}, \quad t_{i,j} \neq 0, \quad i, j = 1, 2, \cdots, n$$

$$s_{ij} \geq 0$$

$$x_1 = 0, x_i \geq 0, \quad i = 1, 2, \cdots, n$$



- 当**sij>0**时，说明对应的作业的开始时间可以推迟**sij**，从而得到每个作业**i**的最迟开工时间。
- 关键路径还可以看成最长路，用求最短路径的方法来求解。

网络最优化模型转化为线性规划模型

本节将某些网络最优化问题转化为线性规划问题来求解，但这并不意味着线性规划的方法是解决这些网络最优化问题的最有效算法.

事实上，这些网络最优化问题均各自存在更有效的算法. 我们之所以将它们转化为线性规划问题，主要基于以下两个方面的考虑：

(1) 我们有现成的功能很强的软件, 它可以方便地求解线性规划问题.

(2) 将某些网络最优化模型, 转化为线性规划模型, 本身就是一种非常好的建模训练, 具体的转化技巧具有重要的启发意义.

PT(Potentialtask graph)图

在**PT(Potentialtask graph)**图中, 用结点表示工序, 如果工序 i 完成之后工序 j 才能启动, 则图中有一条有向边(i, j), 其长度w_i表示工序 i 所需的时间.

这种图必定不存在有向回路, 否则某些工序将在自身完成之后才能开始, 这显然不符合实际情况.

在**PT**图中增加两个虚拟结点v_0和v_n, 使所有仅为始点的结点都直接与v_0联结, v_0为新增边的始点, 这些新增边的权都设为**0**; 使所有仅为终点的结点都直接与v_n联结, v_n为新增边的终点. 这样得到的图G仍然不存在有向回路.

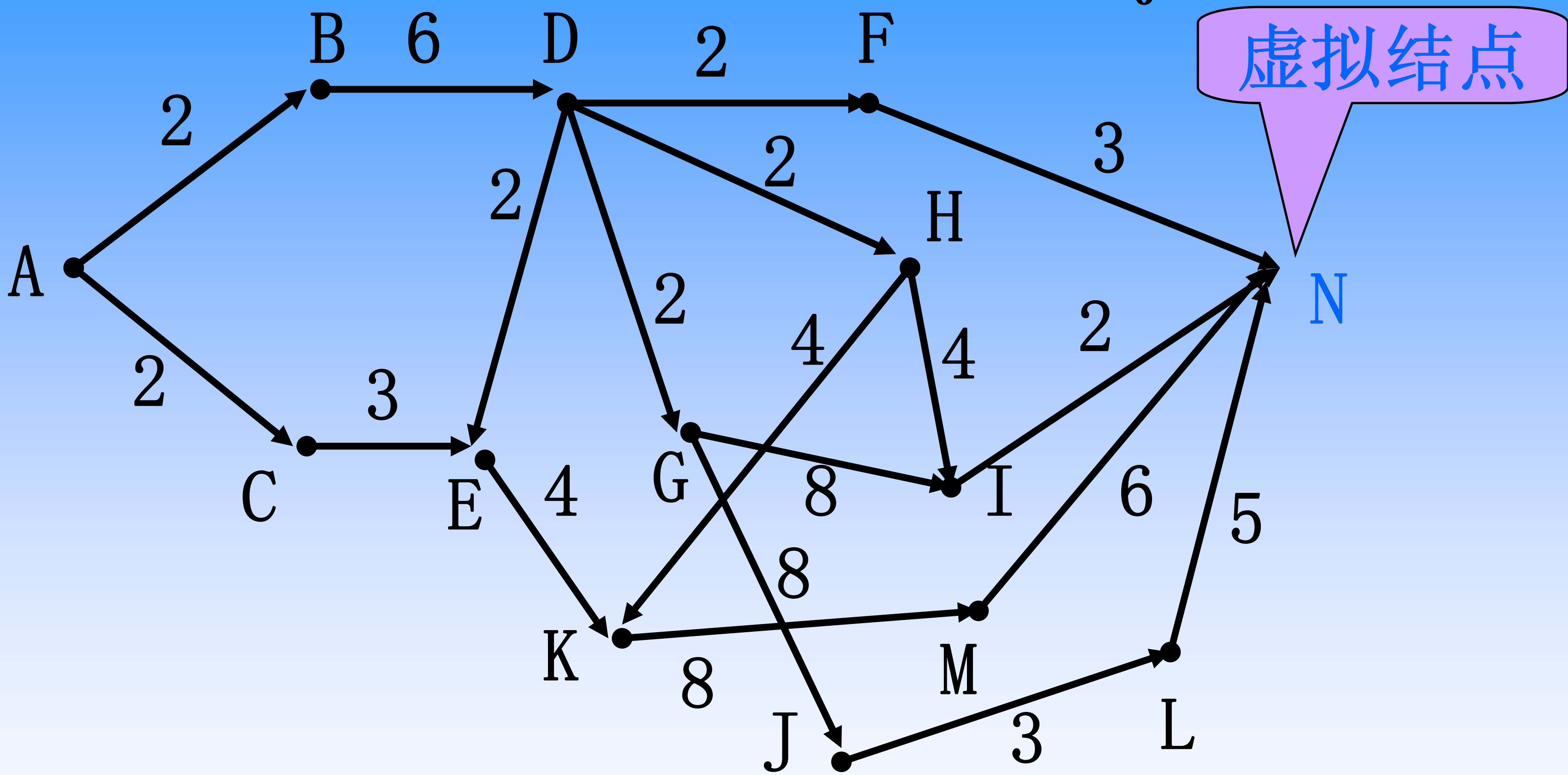
例 一项工程由13道工序组成，所需时间(单位：天)及先行工序如下表所示

工序序号	A	B	C	D	E	F	G	H	I
所需时间	2	6	3	2	4	3	8	4	2
先行工序	—	A	A	B	C,D	D	D	D	G,H

工序序号	J	K	L	M
所需时间	3	8	5	6
先行工序	G	H,E	J	K

试问这项工程至少需要多少天才能完成？那些工程不能延误？那些工程可以延误？最多可延误多少天？

先作出该工程的**PT**图. 由于除了工序A外，均有先行工序, 因此不必虚设虚拟结点v_0.



在**PT**图中,容易看出各工序先后完成的顺序及时间.

关键路径(最长路径)算法

定理 若有向图G中不存在有向回路, 则可以将G的结点重新编号为$u_1, u_2, \ldots, u_n$, 使得对任意的边$u_iu_j \in E(G)$, 都有$i < j$.

各工序最早启动时间算法步骤:

① 根据定理对结点重新编号为$u_1, u_2, \ldots, u_n$.

② 赋初值 $\pi(u_1) = 0$.

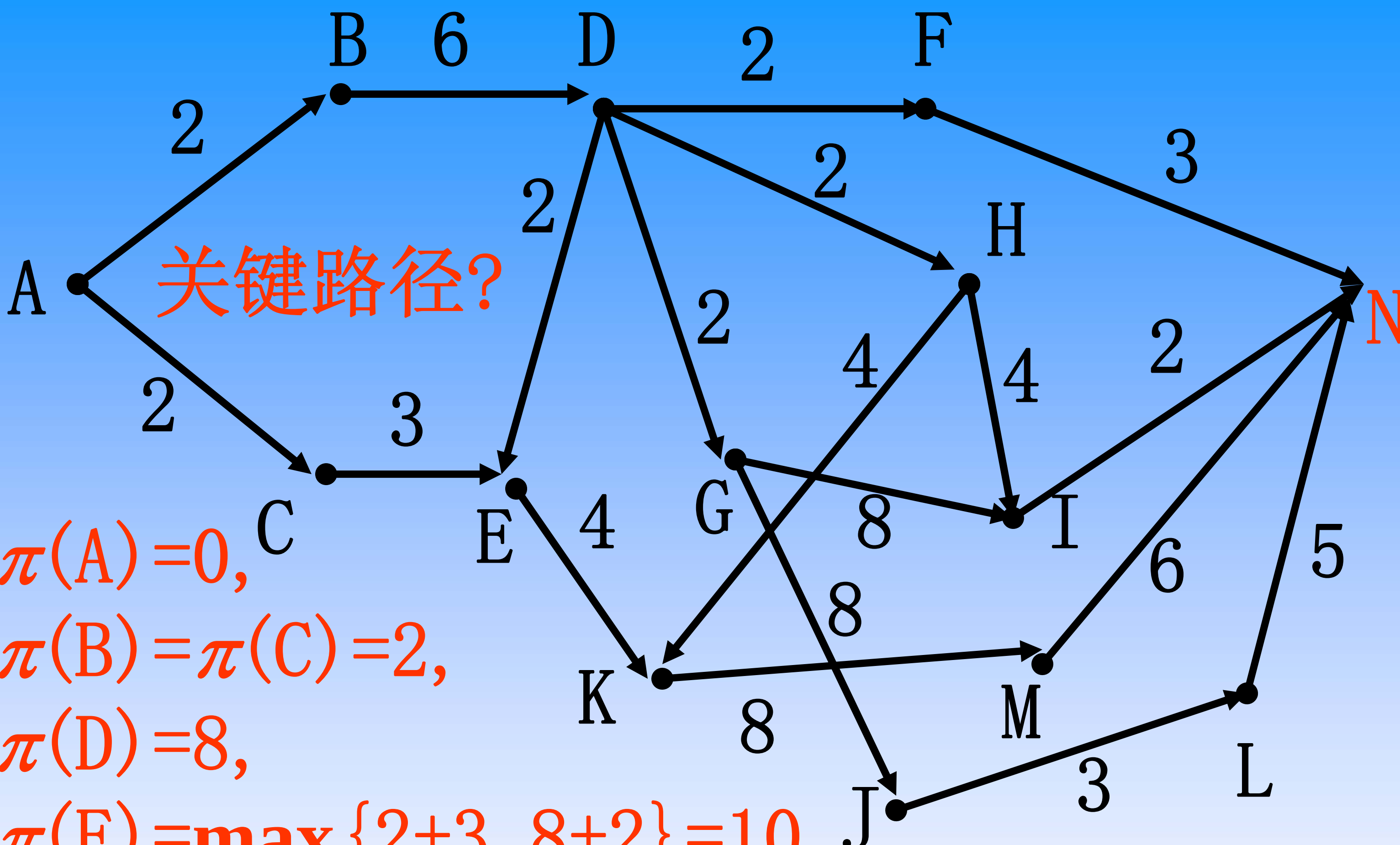
③ 依次更新 $\pi(u_j), j = 2, 3, \ldots, n$.

$$\pi(u_j) = \max\{\pi(u_i) + \omega(u_i, u_j) \mid u_iu_j \in E(G)\}.$$

④ 结束.

其中$\pi(u_j)$表示工序u_j最早启动时间, 而$\pi(u_n)$即$\pi(v_n)$是整个工程完工所需的最短时间.

关键路径?



此例不必重新编号，只要按字母顺序即可.

$\pi(A)=0$,

$\pi(B)=\pi(C)=2$,

$\pi(D)=8$,

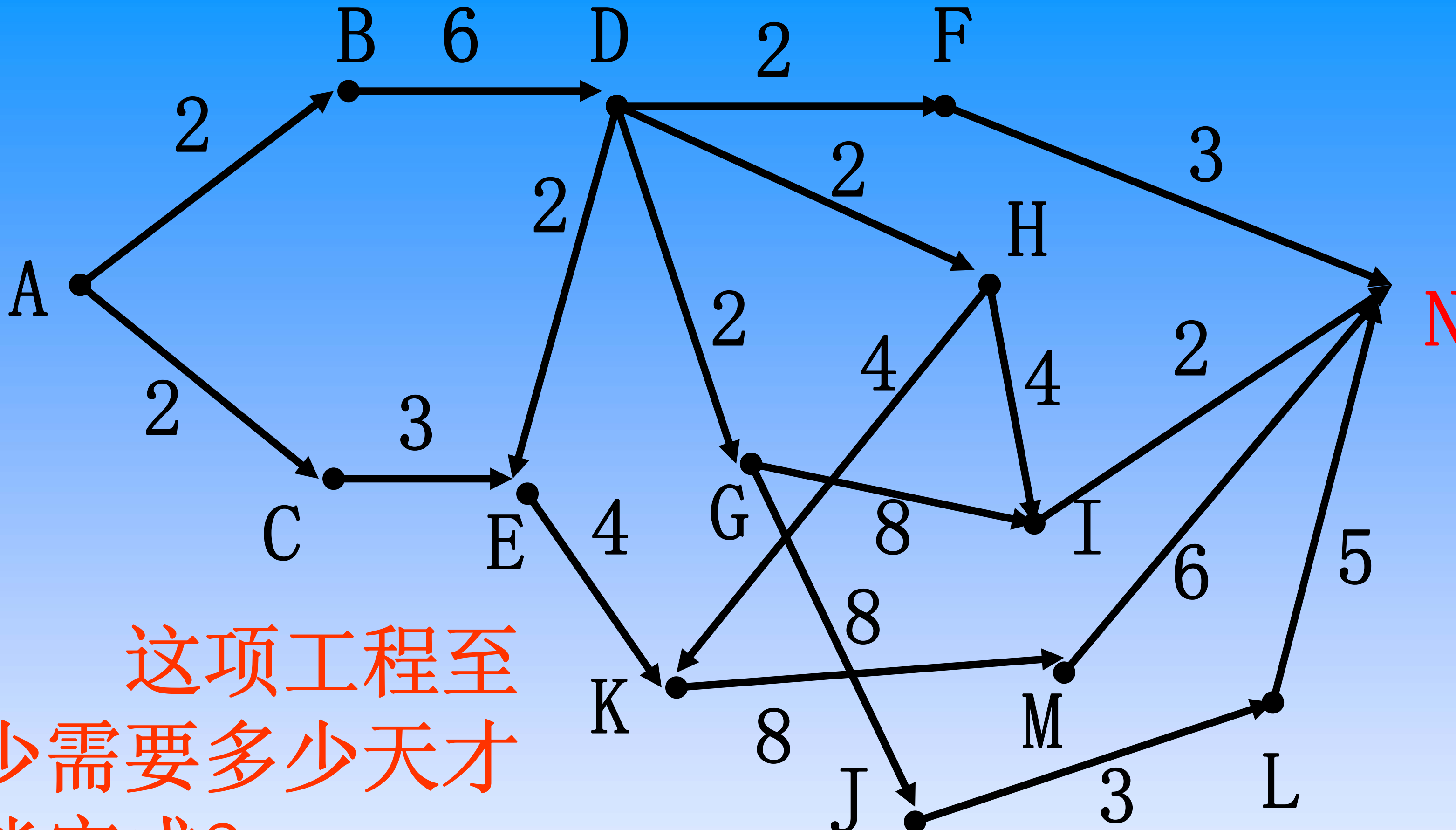
$\pi(E)=\max\{2+3, 8+2\}=10$,

$\pi(F)=\pi(G)=\pi(H)=\pi(D)+2=10$, $\pi(J)=\pi(G)+8=18$,

$\pi(I)=\max\{\pi(G)+8, \pi(H)+4\}=18$, $\pi(L)=\pi(J)+3=21$,

$\pi(K)=\max\{\pi(E)+4, \pi(H)+4\}=14$, $\pi(M)=\pi(K)+8=22$,

$\pi(N)=\max\{\pi(F)+3, \pi(I)+2, \pi(L)+5, \pi(M)+6\}=28$.



这项工程至少需要多少天才能完成?

就是要求A到N的最长路,此路径称为关键路径.

那些工程不能延误? 那些工程可以延误? 最多可延误多少天?

关键路径上的那些工程不能延误.

通过以上计算表明:

这项工程至少需要28天才能完成.

关键路径(最长路径):

$A \to B \to D \to E \to K \to M \to N$

$A \to B \to D \to H \to K \to M \to N$

工序A, B, D, E, H, K, M不能延误, 否则将影响工程的完成.

但是对于不在关键路径上的工序， 是否允许延误？ 如果允许， 最多能够延误多长时间呢？

各工序允许延误时间$t(u_j)$等于各工序最晚启动时间$\tau(u_j)$减去各工序最早启动时间$\pi(u_j)$.

即 $t(u_j)=\tau(u_j)-\pi(u_j)$.