**建模相关**

1.二分图最大（带权）独立集

　　定义：在一个二分图中，每个点都有一个权值，在保证选中的点不相交的情况下，求选出的点的最大权值和。

　　建模：S向X连容量为权值的边，Y向T连容量为权值的边。X与相连的Y连容量为INF的边。

　　原理：求出原图最小割，若原图中存在一条边X->Y，则S->X与Y->T必有一条被割掉，被割边所连的点不选，ans=点权总和-最大流。

2.二分图最小点（权）覆盖

　　定义：假如选了一个点就相当于覆盖了以它为端点的所有边，最小顶点覆盖就是选择最少的点（最小的点权和）来覆盖所有的边。

　　建模：S向X连容量为权值的边，Y向T连容量为权值的边。X与相连的Y连容量为INF的边。

　　原理：求出原图最小割，对原图中每一条边X->Y，S->X与Y->T必有一条被割掉，选择被割边所连的点，该点覆盖了边X->Y。ans=最大流。

3.二分图最小边覆盖

　　定义：假如选了一条边就相当于覆盖了它所连接的两个顶点，最小边覆盖就是选择最少的边来覆盖所有的点。

　　建模：S向X连容量为1的边，Y向T连容量为1的边。X与相连的Y连容量为1的边。

　　原理：实质上是求出原图的最大匹配，若总点数为n，最大匹配为m，则未匹配的(n-2m)个点都只能单独选一条边来覆盖，ans=m+(n-2m)=n-m=总点数-最大流。

4.DAG最小路径覆盖

　　定义：用最少的路径数（要求路径不相交，若路径可相交应先用floyd求出传递闭包）覆盖所有顶点。

　　建模：拆点，S向X连容量为1的边，X'向T连容量为1的边。对每条边X->Y ，由X向Y′ 连一条容量为1的边。

　　原理：每选择一条边X->Y，相当于将以X为终点的路径与以Y为起点的路径合并，路径数就减1。ans=总点数-最大流。

5.最大权闭合图

　　定义：闭合图内的任意点的任意后继也一定在闭合图中，最大权闭合图是一个点权之和最大的闭合图。

　　建模：S向每个正权点连容量为点权的边，每个负权点向T连容量为|点权|的边，对每条边X->Y ，由X向Y 连一条容量为inf的边。

　　原理：（详见 胡伯涛 《最小割模型在信息学竞赛中的应用》）ans=正权点之和-最大流。选择的点与S在同一个割集中，dinic结束后dep!=0的点即为所选点。

**最大流相关**

1.将边(x,y)的流量退回

　　方法：T到y求一次最大流，x到S求一次最大流。

2.在bfs的时候碰到T就返回（玄学优化？）

3.【餐巾计划】（流量可在花费一定费用后重复使用）第i天需要ai流量，连(S,i',ai)和(i,T,ai)，第i天流量的所有来源->i，i'->所有去向。

4.【志愿者招募】（流量只在一段区间内使用且费用不重复计算）第i天需要ai流量，连(i,i',inf-ai)和(i',i+1,inf)，对流量可使用的一段区间[X,Y]，连(S,X,v)和(Y',T,v)。（原理：优先流inf-ai的零权边，缺少的ai会用带权边补齐）

5.无向图中找一条S->T->S的不相交路径相当于找两条S->T的路径

6.【最长k可重区间集问题】将费用取反可以使流优先通过带权/零权边。此问题也可以看成k条不相交路径最大权和。

**最小割相关**

1.判断边(x,y,f)是否可以在最小割中

　　方法1：在残量网络中bfs看是否存在x到y的路径，若不存在则该边可出现在最小割中。

　　方法2：对残量网络缩点，若边(x,y,f)满流并且x和y属于不同SCC，则该边可出现在最小割中。

2.判断边(x,y,f)是否一定在最小割中

　　方法：对残量网络缩点，若边(x,y,f)满流并且S与x属于同一个SCC，y与T属于同一个SCC，则该边一定在最小割中。