1、无源汇有上下界最大流

 题目链接： [sgu194 Reactor Cooling](http://acm.hust.edu.cn/vjudge/problem/viewProblem.action?id=20757)

 题目大意：给n个点，及m根pipe，每根pipe用来流躺液体的，单向的，每时每刻每根pipe流进来的物质要等于流出去的物质，要使得m条pipe组成一个循环体，里面流躺物质。并且满足每根pipe一定的流量限制，范围为[Li,Ri].即要满足每时刻流进来的不能超过Ri(最大流问题)，同时最小不能低于Li。

解题思路：O（-1）。

 建图模型： 以前写的最大流默认的下界为0，而这里的下界却不为0，所以我们要进行再构造让每条边的下界为0，这样做是为了方便处理。对于每根管子有一个上界容量up和一个下界容量low，我们让这根管子的容量下界变为0，上界为up-low。可是这样做了的话流量就不守恒了，为了再次满足流量守恒，即每个节点"入流=出流”，我们增设一个超级源点st和一个超级终点sd。我们开设一个数组du[]来记录每个节点的流量情况。

du[i]=in[i]（i节点所有入流下界之和）-out[i]（i节点所有出流下界之和）。

当du[i]大于0的时候，st到i连一条流量为du[i]的边。

当du[i]小于0的时候，i到sd连一条流量为-du[i]的边。

最后对（st，sd）求一次最大流即可，当所有附加边全部满流时（即maxflow==所有du[]>0之和），有可行解。

2、有源汇有上下界的最大流

题目链接：[zoj3229 Shoot the Bullet](http://acm.hust.edu.cn/vjudge/problem/viewProblem.action?id=20756)

题目大意：一个屌丝给m个女神拍照，计划拍照n天，每一天屌丝最多个C个女神拍照，每天拍照数不能超过D张，而且给每个女神i拍照有数量限制[Li，Ri]，对于每个女神n天的拍照总和不能超过Gi，如果有解求屌丝最多能拍多少张照，并求每天给对应女神拍多少张照；否则输出-1。

解题思路：增设一源点st，汇点sd，st到第i天连一条上界为Di下界为0的边，每个女神到汇点连一条下界为Gi上界为oo的边，对于每一天，当天到第i个女孩连一条[Li，Ri]的边。

建图模型：源点s，终点d。超级源点ss，超级终点dd。首先判断是否存在满足所有边上下界的可行流，方法可以转化成无源汇有上下界的可行流问题。怎么转换呢？

增设一条从d到s没有下界容量为无穷的边，那么原图就变成了一个无源汇的循环流图。接下来的事情一样，超级源点ss连i（du[i]>0），i连超级汇点（du[i]<0）,

对（ss，dd）进行一次最大流，当maxflow等于所有(du[]>0)之和时，有可行流，否则没有。

当有可行流时，删除超级源点ss和超级终点dd，再对（s，d）进行一次最大流，此时得到的maxflow则为题目的解。为什么呢？因为第一次maxflow（）只是求得所有满足下界的流量，而残留网络（s，d）路上还有许多自由流（没有和超级源点和超级汇点连接的边）没有流满，所有最终得到的maxflow=（第一次流满下界的流+第二次能流通的自由流）。

3、有源汇有上下界的最小流

题目链接：   sgu176  [Flow construction](http://acm.sgu.ru/problem.php?contest=0&problem=176)

题目大意：有一个类似于工业加工生产的机器，起点为1终点为n，中间生产环节有货物加工数量限制，输出u v z c， 当c等于1时表示这个加工的环节必须对纽带上的货物全部加工（即上下界都为z），c等于0表示加工没有上界限制，下界为0，求节点1（起点）最少需要投放多少货物才能传送带正常工作。

解题思路：

   1、du[i]表示i节点的入流之和与出流之和的差。

   2、增设超级源点st和超级汇点sd，连（st，du[i]（为正）），（-du[i]（为负），sd）。 ///增设超级源点和超级汇点，因为网络中规定不能有弧指向st，也不能有流量流出sd

   3、做一次maxflow（）。

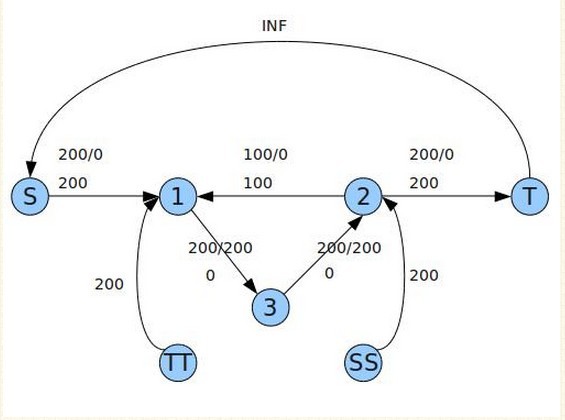
   4、源点（Sd）和起点（St）连一条容量为oo的边。

   5、再做一次maxflow（）。

   6、当且仅当所有附加弧满载时有可行流，最后答案为flow[（Sd->St）^1]，St到Sd最大流就是Sd到St最小流。

建图模型：同样转换成先求无源汇有上下界的可行流，先添加一条d到s容量为无穷的边，这里求最小流很容易让人产生歧路，为什么呢？当所有边满足下界条件并且能量守恒时，这时候求得的最大流不就是最小流么。这样是错误了，我开始了在这揣测了良久。

下面来看个例子：



这样求得的最小流为200，而实际的可行最小流解只需100。

问题出在原图中存在环（循环流），而我们没有利用，导致流增大了。

解决方法：先不增加d->s容量为无穷的边，进行一次maxflow（），如果还没有满流，则加一条（d，s）容量为无穷的边，再进行一次maxflow（），当且仅当所有附加弧满载时，有可行解，解为flow[（d->s）^1]（即d到s的后悔边权值）。