图论 其他算法

华南理工大学 江熠玲

2020 年 2 月 2 日

前言

前言

今天讲课的内容相对基础，当作巩固。

课件中题目若未特殊声明，时限1s，内存128M。

欢迎踊跃回答。

引言：基本概念

引言：基本概念

图论〔Graph Theory〕是数学的一个分支。它以图为研究对象。图论中的图是由若干给定的点及连接两点的线所构成的图形，这种图形通常用来描述某些事物之间的某种特定关系，用点代表事物，用连接两点的线表示相应两个事物间具有这种关系。

目录

目录

一笔画问题

一笔画问题

如果一个图存在一笔画，则一笔画的路径叫做**欧拉路**，如果最后又回到起点，那这个路径叫做**欧拉回路**。

存在欧拉回路的图称为**欧拉图**，存在欧拉路径但不存在欧拉回路的图称为**半欧拉图**。

奇点：跟这个点相连的边数目有奇数个的点。

一笔画问题

一笔画问题

存在**欧拉路**的条件：图是连通的，有且只有2 个奇点。

存在**欧拉回路**的条件：图是连通的，有0 个奇点。

一笔画问题

存在**欧拉路**的条件：图是连通的，有且只有2 个奇点。

存在**欧拉回路**的条件：图是连通的，有0 个奇点。

两个定理的正确性是显而易见的，既然每条边都要经过一次，那么对于欧拉路，除了起点和终点外，每个点如果进入了一次，显然一定要出去一次，显然是偶点。对于欧拉回路，每个点进入和出去次数一定都是相等的，显然没有奇点。

一笔画问题

一笔画问题

如果寻找欧拉回路，对任意一个点进行dfs，寻找欧拉路，则对一个度为1 的点进行dfs，时间复杂度为$O(m+n)$，m为边数，n 为点数。

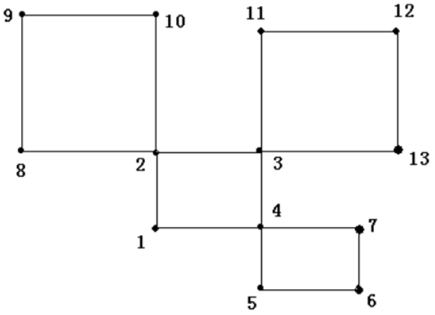
一笔画问题

一笔画问题

欧拉回路**局限性**：

一笔画问题

欧拉回路**局限性：**只能找出一个欧拉回路。



一笔画问题的衍伸：哈密尔顿环

一笔画问题的衍伸：哈密尔顿环

欧拉回路是指不重复地走过**所有的路径**并且最后还能回到起点的回路。

一笔画问题的衍伸：哈密尔顿环

欧拉回路是指不重复地走过**所有的路径**并且最后还能回到起点的回路。

哈密尔顿环是指不重复地走过**所有的点**并且最后还能回到起点的回路。

Hierholzer算法

Hierholzer算法

步骤：

1. 判断奇点数。奇点数若为0则任意指定起点，奇点数若为2则指定起点为奇点。
2. 开始递归函数Hierholzer(x):

    循环寻找与x相连的边(x,u):

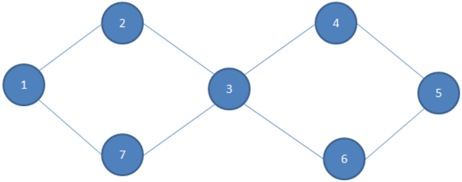
        删除(x,u)

        删除(u,x)

        Hierholzer(u);

    将x插入答案队列之中

Hierholzer算法



找到该图没有奇点，从1开始。删边1-2 递归到2;删边2-3 递归到3;删边3-7 递归到7 删边7-1 递归到1;1无边，1加入队列，返回;7加入队列，返回;

删边3-4 递归到4;删边4-5 递归到5;删边5-6 递归到6;删边6-3 递归到3;3加入队列，返回; 6,5,4,3,2,1依次加入队列，返回

答案队列为：1 7 3 6 5 4 3 2 1。反向输出即为答案。

Hierholzer算法

Hierholzer算法

```
void euler(int u)
{
    for(int i = bg; i <= ed; i++)
        if (m[u][i] >= 1)
        {
            m[u][i]--;
            m[i][u]--;
            euler(i);
        }
    ans[++anssize] = u;
}
```

拓扑排序

拓扑排序

现在来考虑一个问题：现在有一个工程，这个工程被分成了很多部分。有一些部分要求前面某些部分完成后才可以开始进行。有些部分则可以同时进行。

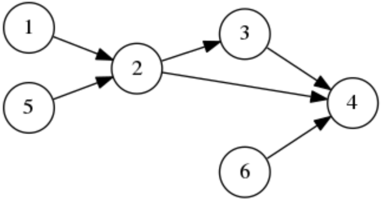
拓扑排序

现在来考虑一个问题：现在有一个工程，这个工程被分成了很多部分。有一些部分要求前面某些部分完成后才可以开始进行。有些部分则可以同时进行。

我们可以把每个部分看作一个结点，刚刚这些限制看成是边。

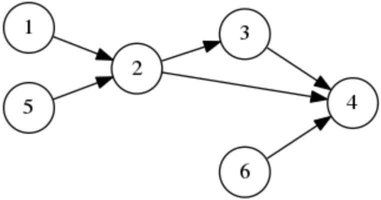
拓扑排序

拓扑排序



比如说这张图就可以看成是这样一个限制。这样的图有个特点：没有环。

拓扑排序



比如说这张图就可以看成是这样一个限制。这样的图有个特点：没有环。

因此这样的图也被成为**DAG（有向无环图）**。

求出一个这个工程的工作序列的算法被称为**拓扑排序**。

拓扑排序

拓扑排序

拓扑排序的过程大概是这样的：

1. 选择一个入度为0 的结点并直接输出。

拓扑排序

拓扑排序的过程大概是这样的：

① 选择一个入度为0 的结点并直接输出。

② 删除这个结点以及与它关联的所有边。

拓扑排序

拓扑排序的过程大概是这样的：

① 选择一个入度为0 的结点并直接输出。

② 删除这个结点以及与它关联的所有边。

③ 重复步骤①和②，直到找不到入度为0 的结点。

拓扑排序的过程大概是这样的：

① 选择一个入度为0 的结点并直接输出。

② 删除这个结点以及与它关联的所有边。

③ 重复步骤①和②，直到找不到入度为0 的结点。

通常情况下，在实现的时候会维护一个队列以及每个结点的入度。在删除边的时候顺便把相应结点的入度减去，当这个结点入度为0 的时候直接将其加入队列。

拓扑排序

拓扑排序

```
void topo()
{
    int i,j,now;
    queue<int> q;
    for(i=1;i<=n;i++)
        if(!ind[i]) q.push(i);
    while(!q.empty())
    {
        now=q.front();q.pop();
        ans[++cnt]=now;
        for(i=last[now];i;i=e[i].next)
        {
            ind[e[i].to]--;
            if(!ind[e[i].to]) q.push(e[i].to);
        }
    }
    if(cnt!=n) puts("-1");
    else for(i=1;i<=n;i++) printf("%d ",ans[i]);

}
```

差分约束系统

差分约束系统

差分约束系统（system of difference constraints），是求解关于一组变数的特殊不等式组之方法。如果一个系统由n个变量和m 个约束条件组成，其中每个约束条件形如$x_j - x_i \leq b_k (i, j \in [1, n], k \in [1, m])$,则称其为差分约束系统(system of difference constraints)。差分约束系统是求解关于一组变量的特殊不等式组的方法。

通俗一点地说，差分约束系统就是一些不等式的组，而我们的目标是通过给定的约束不等式组求出最大值或者最小值或者差分约束系统是否有解。

差分约束系统

差分约束系统

之所以差分约束系统可以通过图论的最短路来解，是因为$x_j - x_i <= b_k$类似最短路中的三角不等式$d[v] \leq d[u] + w[u][v]$，即$d[v] - d[u] \leq w[u][v]$。而求取最大值的过程类似于最短路算法中的松弛过程。

差分约束系统

之所以差分约束系统可以通过图论的最短路来解，是因为$x_j - x_i <= b_k$类似最短路中的三角不等式$d[v] \leq d[u] + w[u][v]$，即$d[v] - d[u] \leq w[u][v]$。而求取最大值的过程类似于最短路算法中的松弛过程。

三角不等式：

$B - A \leq c$

$C - B \leq a$

$C - A \leq b$

差分约束系统

之所以差分约束系统可以通过图论的最短路来解，是因为$x_j - x_i <= b_k$类似最短路中的三角不等式$d[v] \leq d[u] + w[u][v]$，即$d[v] - d[u] \leq w[u][v]$。而求取最大值的过程类似于最短路算法中的松弛过程。

三角不等式：

$B - A \leq c$

$C - B \leq a$

$C - A \leq b$

如果要求$C - A$的最大值，可以知道为$min(b, a + c)$,而这正对应了C到A的最短路。

差分约束系统

差分约束系统

对三角不等式加以推广，变量n个，不等式m个，要求$x_n - x_1$的最大值，便就是求取建图后的最短路。

差分约束系统

对三角不等式加以推广，变量n个，不等式m个，要求$x_n - x_1$的最大值，便就是求取建图后的最短路。

同样地，如果要求取差分约束系统中$x_n - x_1$的最小值，便是求取建图后的最长路。

差分约束系统

对三角不等式加以推广，变量n个，不等式m个，要求$x_n - x_1$的最大值，便就是求取建图后的最短路。

同样地，如果要求取差分约束系统中$x_n - x_1$的最小值，便是求取建图后的最长路。

注意，建图后不一定存在最短路/最长路，因为可能存在无限减小/增大的负环/正环，题目一般会对应于不同的输出。判断差分约束系统是否存在解一般判环即可。

差分约束系统

差分约束系统

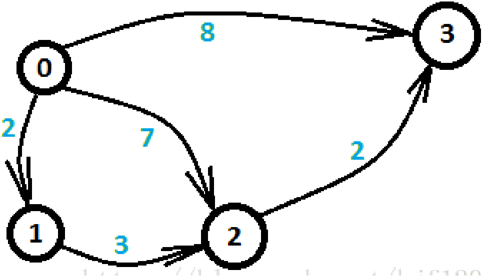
$x1 - x0 <= 2$ (1)

$x2 - x0 <= 7$ (2)

$x3 - x0 <= 8$ (3)

$x2 - x1 <= 3$ (4)

$x3 - x2 <= 2$ (5)



目录

目录

例题

例题

[HDU 1285]确定比赛名次

[HDU 1285]确定比赛名次

有N个比赛队，编号依次为$1, 2, \cdots, N$进行比赛，比赛结束后，裁判委员会要将所有参赛队伍从前往后依次排名，但现在裁判委员会不能直接获得每个队的比赛成绩，只知道每场比赛的结果，即P_1赢P_2，用P_1, P_2表示排名时P_1在P_2之前。现在请你编程序确定排名。

$1 \leq N \leq 500$。

[HDU 1285]确定比赛名次

[HDU 1285]确定比赛名次

拓扑排序。

[POJ 1386]Play on Words

[POJ 1386]Play on Words

给你若干个字符串，一个单词的尾部和一个单词的头部相同那么这两个单词就可以相连，判断给出的n个单词是否能够一个接着一个全部连通。

$1 \leq n \leq 10^5$。

[POJ 1386]Play on Words

[POJ 1386]Play on Words

对于每一个单词，我们只需要纪录下它的首字母和尾字母即可，然后单词接龙就变为了我们熟知的欧拉路径问题了。

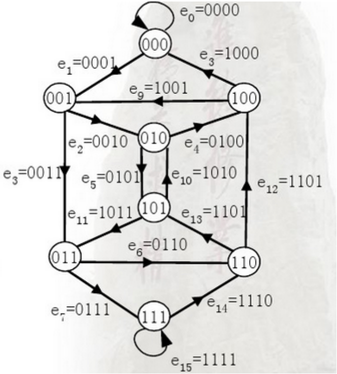
有向图中的欧拉图判定条件:

1. 所有的点联通
2. 欧拉回路中所有点的入度和出度一样.
3. 欧拉通路中终点的入度-出度=1，起点的出度-入度=1，其他的所有点入度=出度;

给出k，求一个最长的M位01串，使其从每一个位置向后走k个得到的M个k位01串互不相同（最后一个和第一个相邻，即是一个环）。输出字典序最小的答案。

$2 \leq k \leq 11$。

第二问我们将每个k二进制数看成一个点，在它后面加上0/1就能得到两个新数，我们从这个数向两个新数连边，于是这就变成了求一个欧拉回路的问题。显然此题是存在欧拉回路的第一问答案为2^k,第二问暴力求欧拉回路即可。



[HDU 5883]The Best Path

[HDU 5883]The Best Path

给你一个n个点m条边的无向图，每个点有权值，你要从某一个点出发，使得一笔画经过所有的路，且使得经过的节点的权值XOR运算最大，求最大值，或输出*impossible*。

[HDU 5883]The Best Path

[HDU 5883]The Best Path

首先得知道一些关于XOR运算的性质：

1. 如果一个数异或偶数次，那么这个结果是0。
2. 如果一个序列是确定的，那么异或的顺序不影响最终的结果。

[HDU 5883]The Best Path

首先得知道一些关于XOR运算的性质：

1. 如果一个数异或偶数次，那么这个结果是0。
2. 如果一个序列是确定的，那么异或的顺序不影响最终的结果。

有了这样的性质以后，这道题就很简单了。

1. 它必须是联通的，对于这个题目，如果发现的边有很多联通快，这肯定是不合法的。
2. 其次，计算每个点的度数，如果奇点的个数不为0 并且不为2，那么也是不合法。
3. 然后只要知道每个点的度数就可以算了，不用dfs 去模拟路。

[LUOGU 1341]无序字母对

[LUOGU 1341]无序字母对

给定n个各不相同的无序字母对（区分大小写，无序即字母对中的两个字母可以位置颠倒）。请构造一个有$n+1$个字母的字符串使得每个字母对都在这个字符串中出现。

[LUOGU 1341]无序字母对

[LUOGU 1341]无序字母对

每个相邻字母连无向边。

[LUOGU 1341]无序字母对

每个相邻字母连无向边。

欧拉路径。

[Codeforces 516B]Drazil and Tiles

[Codeforces 516B]Drazil and Tiles

给一个$n \times m$的矩阵，问在其中空白处放置$1 \times 2, 2 \times 1$的骨牌的方案是否存在且唯一。

$1 \leq n, m \leq 1000$。

[Codeforces 516B]Drazil and Tiles

[Codeforces 516B]Drazil and Tiles

我们这里定义一个顶点的度就是其周围点的个数。

[Codeforces 516B]Drazil and Tiles

我们这里定义一个顶点的度就是其周围点的个数。

那么对应度为1的点是一定确定了放置这个点的方式的，对应这个点的四周四个位子中，此时一定只有一个点，那么对应将此处和那个位子直接放置这个1×2的形状即可。

[Codeforces 516B]Drazil and Tiles

我们这里定义一个顶点的度就是其周围点的个数。

那么对应度为1的点是一定确定了放置这个点的方式的，对应这个点的四周四个位子中，此时一定只有一个点，那么对应将此处和那个位子直接放置这个1×2的形状即可。

然后处理完这个点之后，对其他点进行操作，如果再发现了度为1的点，入队。一直处理下去这样的类拓扑排序的过程即可。

[Codeforces 516B]Drazil and Tiles

我们这里定义一个顶点的度就是其周围点的个数。

那么对应度为1的点是一定确定了放置这个点的方式的，对应这个点的四周四个位子中，此时一定只有一个点，那么对应将此处和那个位子直接放置这个1×2的形状即可。

然后处理完这个点之后，对其他点进行操作，如果再发现了度为1的点，入队。一直处理下去这样的类拓扑排序的过程即可。

如果最后还剩下了，说明无法完全覆盖，或者方案不止一个。

混合图

混合图

你有一个混合图，它有n 个顶点，m 条边，其中a 条是有向边，b 条是无向边。现在要求为这b 条无向边确定一个方向，使得最后的有向图上不存在环。

$1 \le n, a, b \le 10^5$，$m = a + b$。

不用考虑无解的情况。

混合图

混合图

既然要求最后的有向图没有环，也就是要求最后是个DAG。

混合图

既然要求最后的有向图没有环，也就是要求最后是个*DAG*。

DAG 进行拓扑排序后边的方向都是拓扑序小的连向拓扑序大的。

既然要求最后的有向图没有环，也就是要求最后是个*DAG*。

DAG 进行拓扑排序后边的方向都是拓扑序小的连向拓扑序大的。

先把无向边忽略，对图进行拓扑排序，最后无向边的方向就是拓扑序小的连向拓扑序大的。

[POJ 3159]Candies

[POJ 3159]Candies

有n 个人分糖果，现在有m 个限制，比如说某一个人得到的糖果不能比另一个人少k 个以上。

请你求出小A 比小B 最多能多得到多少个糖果。

其中$1 \leq n \leq 30000, 1 \leq m \leq 150000$。

Candies

Candies

差分约束系统。

比如小B 得到的糖果不能比小A 少k 个以上，那么

$$A - B \leq K$$

Candies

差分约束系统。

比如小B 得到的糖果不能比小A 少k 个以上，那么

$$A - B \leq K$$

将每个人看成一个点，比如上面这个限制就从B 向A 连接一条长度为k 的边。

跑一遍最短路即可。

[POJ 3169]Layout

[POJ 3169]Layout

有n头牛，他们按顺序排成了一排，有些牛关系比较好，他们的距离不能超过某个距离，还有些牛关系不好，他们之间的距离不能小于某个距离，可能会有多头牛挤在同一位置上，问1号牛和n号牛之间的最大距离是多少，如果不存在满足条件的排列则输出-1，如果距离无限大则输出-2。

$2 \leq n \leq 1000$。

[POJ 3169]Layout

[POJ 3169]Layout

令d[i]表示第i头牛的位置，因为牛按顺序排列，则有$d[i] \leq d[i+1]$，即$d[i] - d[i+1] \leq 0$。

[POJ 3169]Layout

令d[i]表示第i头牛的位置，因为牛按顺序排列，则有$d[i] \leq d[i+1]$，即$d[i] - d[i+1] \leq 0$。

关系不好的牛有$d[a] + d \leq d[b]$。

[POJ 3169]Layout

令d[i]表示第i头牛的位置，因为牛按顺序排列，则有$d[i] \leq d[i+1]$，即$d[i] - d[i+1] \leq 0$。

关系不好的牛有$d[a] + d \leq d[b]$。

关系好的牛有$d[a] + D \leq d[b]$,

[POJ 3169]Layout

令d[i]表示第i头牛的位置，因为牛按顺序排列，则有$d[i] \le d[i+1]$，即$d[i] - d[i+1] \le 0$。

关系不好的牛有$d[a] + d \le d[b]$。

关系好的牛有$d[a] + D \le d[b]$, 即$d[a] - d[n] \le -D$ 。

跑最短路即可。

[POJ 2983]Is the Information Reliable?

[POJ 2983]Is the Information Reliable?

银河系中有n个防御站排成一条直线，给定m条信息，每条信息的格式如下:

$P\ A\ B\ X$代表A站在B站北方X光年处。

$V\ A\ B$ 代表A站在B站北方，而且距离至少为1光年。

问是否存在这样的一个防御战排列满足上述要求，是输出Reliable，否输出Unreliable。

[POJ 2983]Is the Information Reliable?

[POJ 2983]Is the Information Reliable?

差分约束系统。

[POJ 2983]Is the Information Reliable?

差分约束系统。

$dist[A] - dist[B] \geq X$，表示A在B北边至少X光年位置。变形为：$dist[B] <= dist[A] - X$;所以A指向B有条权值为-X的边。

[POJ 2983]Is the Information Reliable?

差分约束系统。

$dist[A]-dist[B]\geq X$，表示A在B北边至少X光年位置。变形为：$dist[B] <= dist[A]-X$;所以A指向B有条权值为-X的边。

对于$A-B=X$，

[POJ 2983]Is the Information Reliable?

差分约束系统。

$dist[A] - dist[B] \geq X$，表示A在B北边至少X光年位置。变形为：$dist[B] <= dist[A] - X$;所以A指向B有条权值为-X的边。

对于$A - B = X$，建两条边$dist[B] \leq dist[A] - X$，$dist[A] \leq dist[B] + X$。

[POJ 2983]Is the Information Reliable?

差分约束系统。

$dist[A]-dist[B] \geq X$，表示A在B北边至少X光年位置。变形为：$dist[B] <= dist[A]-X$;所以A指向B有条权值为-X的边。

对于$A-B=X$，建两条边$dist[B] \leq dist[A]-X$，$dist[A] \leq dist[B]+X$。

SPFA存在负权环则说明不符合实际情况。

[HDU 3440]House Man

[HDU 3440]House Man

T组数据，有n栋房子在一条直线上，给出每栋房子的高度和开始时的相对位置，可以移动一些房子，但不能改变这些房子的相对位置，现在从最矮的房子开始，每次跳至第一个比它高的房子，经过$n-1$跳到达最高的房子。而且每次跳跃的水平距离最大是D，房子不能在同一位置，只能在整点位置。问最矮的房子和最高的房子之间的最大距离可以是多少？如果不能从最矮的房子到达最高的房子则输出-1。

$1 \le T \le 500, 1 \le N \le 500$。

[HDU 3440]House Man

[HDU 3440]House Man

令$d[i]$表示第i栋房子与第一栋房子之间的最大距离，那么我们要求的就是的$d[n]$。

[HDU 3440]House Man

令$d[i]$表示第i栋房子与第一栋房子之间的最大距离，那么我们要求的就是的$d[n]$。

首先每栋房子之间的相对位置已经确定且不能在同一位置，那么$d[i+1] > d[i]$，即$d[i+1] - d[i] \geq 1$,即$d[i] - d[i+1] \leq -1$。

令$d[i]$表示第i栋房子与第一栋房子之间的最大距离，那么我们要求的就是的$d[n]$。

首先每栋房子之间的相对位置已经确定且不能在同一位置，那么$d[i+1] > d[i]$，即$d[i+1] - d[i] \geq 1$,即$d[i] - d[i+1] \leq -1$。

还有一个条件是相邻两个要跳跃的距离不可以大于D，可以写出另一个方程。

[HDU 3440]House Man

令$d[i]$表示第i栋房子与第一栋房子之间的最大距离，那么我们要求的就是的$d[n]$。

首先每栋房子之间的相对位置已经确定且不能在同一位置，那么$d[i+1] > d[i]$，即$d[i+1] - d[i] \geq 1$,即$d[i] - d[i+1] \leq -1$。

还有一个条件是相邻两个要跳跃的距离不可以大于D，可以写出另一个方程。

有一点小问题就是建边的方向和跳跃的方向不同，我们要保证是向一个方向建图的。

[HDU 3440]House Man

令$d[i]$表示第i栋房子与第一栋房子之间的最大距离，那么我们要求的就是的$d[n]$。

首先每栋房子之间的相对位置已经确定且不能在同一位置，那么$d[i+1] > d[i]$，即$d[i+1] - d[i] \geq 1$,即$d[i] - d[i+1] \leq -1$。

还有一个条件是相邻两个要跳跃的距离不可以大于D，可以写出另一个方程。

有一点小问题就是建边的方向和跳跃的方向不同，我们要保证是向一个方向建图的。

若从左到右建边，如果1在n的左面就从1到n跑SPFA，反之则从n到1跑最短路了。

[POJ 1275]Cashier Employment

[POJ 1275]Cashier Employment

在一家超市里，每个时刻都需要有营业员看管，$R(i)(0 \leq i < 24)$表示从i时刻开始到$i+1$时刻结束需要的营业员的数目，现在有N 个申请人申请这项工作，并且每个申请者都有一个起始工作时间t_i，如果第i个申请者被录用，那么他会从t_i 时刻开始连续工作8 小时。现在要求选择一些申请者进行录用，使得任何一个时刻i，营业员数目都能大于等于$R(i)$。求出至少需要录用多少营业员。

$N \leq 1000$。

[POJ 1275]Cashier Employment

[POJ 1275]Cashier Employment

每天的工作情况都是一样的，我们只需要求出一天的即可。根据题意，令$s[i]$为一天内前$i+1$个小时录用的人数。

- 如果$i \geq 7$,则$s[i]-s[i-8] \geq R[i]$

[POJ 1275]Cashier Employment

每天的工作情况都是一样的，我们只需要求出一天的即可。根据题意，令$s[i]$为一天内前$i+1$个小时录用的人数。

- 如果$i \geq 7$,则$s[i] - s[i-8] \geq R[i]$
- 如果$0 \leq i < 7$，则可以推出$s[23] - s[i+16] + s[i] \geq R[i]$

[POJ 1275]Cashier Employment

每天的工作情况都是一样的，我们只需要求出一天的即可。根据题意，令$s[i]$为一天内前$i+1$个小时录用的人数。

- 如果$i \geq 7$,则$s[i]-s[i-8] \geq R[i]$
- 如果$0 \leq i < 7$，则可以推出$s[23]-s[i+16]+s[i] \geq R[i]$
- 同时每个时刻录用的人数有个上限，假设第i时刻最多可以录用的人数为$b[i]$,则对于每一个i有$0 \leq s[i]-s[i-1] \leq b[i]$。

[POJ 1275]Cashier Employment

每天的工作情况都是一样的，我们只需要求出一天的即可。根据题意，令$s[i]$为一天内前$i+1$个小时录用的人数。

- 如果$i \geq 7$,则$s[i]-s[i-8] \geq R[i]$
- 如果$0 \leq i < 7$，则可以推出$s[23]-s[i+16]+s[i] \geq R[i]$
- 同时每个时刻录用的人数有个上限，假设第i时刻最多可以录用的人数为$b[i]$,则对于每一个i有$0 \leq s[i]-s[i-1] \leq b[i]$。

第二个不等式中包含3个s数组的变量，这该怎么建图呢？

[POJ 1275]Cashier Employment

每天的工作情况都是一样的，我们只需要求出一天的即可。根据题意，令$s[i]$为一天内前$i+1$个小时录用的人数。

- 如果$i \geq 7$,则$s[i]-s[i-8] \geq R[i]$
- 如果$0 \leq i < 7$，则可以推出$s[23]-s[i+16]+s[i] \geq R[i]$
- 同时每个时刻录用的人数有个上限，假设第i时刻最多可以录用的人数为$b[i]$,则对于每一个i有$0 \leq s[i]-s[i-1] \leq b[i]$。

第二个不等式中包含3个s数组的变量，这该怎么建图呢？

枚举$s[23]$，那么这个量就是已知的了，因此不等式可以变为$s[i]-s[i+16] \geq R[i]-s[23]$。

[POJ 1275]Cashier Employment

每天的工作情况都是一样的，我们只需要求出一天的即可。根据题意，令$s[i]$为一天内前$i+1$个小时录用的人数。

- 如果$i \geq 7$,则$s[i]-s[i-8] \geq R[i]$
- 如果$0 \leq i < 7$，则可以推出$s[23]-s[i+16]+s[i] \geq R[i]$
- 同时每个时刻录用的人数有个上限，假设第i时刻最多可以录用的人数为$b[i]$,则对于每一个i有$0 \leq s[i]-s[i-1] \leq b[i]$。

第二个不等式中包含3个s数组的变量，这该怎么建图呢？

枚举$s[23]$，那么这个量就是已知的了，因此不等式可以变为$s[i]-s[i+16] \geq R[i]-s[23]$。

既然$s[23]$是枚举的一天的录用人数的最小数目，我们建图之后求出的$s[23]$也应该为枚举的那个值。

每天的工作情况都是一样的，我们只需要求出一天的即可。根据题意，令$s[i]$为一天内前$i+1$个小时录用的人数。

- 如果$i \geq 7$,则$s[i]-s[i-8] \geq R[i]$
- 如果$0 \leq i < 7$，则可以推出$s[23]-s[i+16]+s[i] \geq R[i]$
- 同时每个时刻录用的人数有个上限，假设第i时刻最多可以录用的人数为$b[i]$,则对于每一个i有$0 \leq s[i]-s[i-1] \leq b[i]$。

第二个不等式中包含3个s数组的变量，这该怎么建图呢？

枚举$s[23]$，那么这个量就是已知的了，因此不等式可以变为$s[i]-s[i+16] \geq R[i]-s[23]$。

既然$s[23]$是枚举的一天的录用人数的最小数目，我们建图之后求出的$s[23]$也应该为枚举的那个值。单调性，二分。

基础练习

基础练习

结束啦

结束啦

欢迎提问

Questions are welcomed!