**问题背景**

化工企业是个高危行业，设备数量和种类相对较多，生产环环相扣，其中每一个隐患点若不及时处理，不仅严重威胁着人民的生命安全，而且对实现长周期的稳定生产带来制约，影响企业的生产效益。

Chemical enterprises are a high-risk industry, with a relatively large number and types of equipment, and production is interlinked. If each hidden danger is not dealt with in time, it will not only seriously threaten the safety of people's lives, but also restrict the realization of long-term stable production, affecting the production efficiency of enterprises.

显而易见，定期的巡查监视，可以及时发现隐患，使其处理在萌芽状态，从而最大限度地减少该隐患带来的不必要的损失。

Obviously, regular inspection and monitoring can detect hidden dangers in time and make them deal with them in the bud, thereby minimizing the unnecessary losses caused by the hidden dangers.

**问题重述：**

某化工厂共有26个巡检点需要工人进行检查，以保证企业的正常生产。现需要建立相关模型，用于确定巡检人数和规划巡检路线，并且，在尽可能减少人力资源消耗的前提下，保证每一个工人工作量的平衡。

A chemical plant has a total of 26 inspection points that require workers to inspect to ensure the normal production of the enterprise. Now your team requires the establishment of relevant models for determining the number of inspections and planning inspection routes. At the same time, under the premise of reducing the consumption of human resources as much as possible, ensure the balance of each worker's workload.

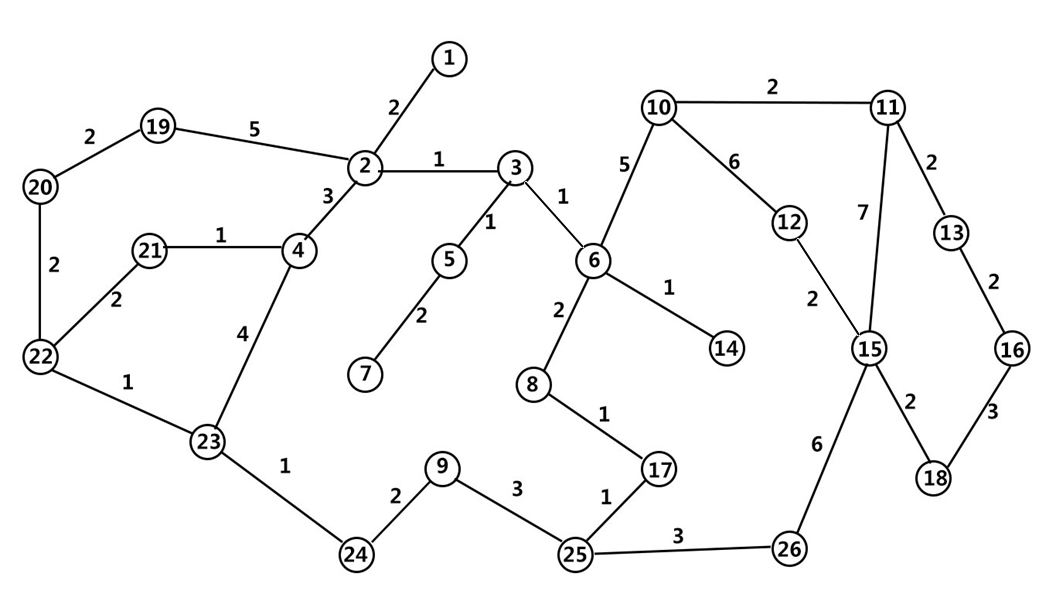
如果上班时间固定，巡检人员中途不休息，每天实行三班倒，每班工作8个小时，则每班至少需要多少人，并给出工人的巡检路线和时间表。

If the working hours are fixed and the inspectors do not rest in the middle of the journey, three shifts are implemented every day, and each shift works for 8 hours. At least how many people are required for each shift, and the inspection routes and timetables of the workers are given.

如果考虑巡检人员每隔2小时需要休息一次，时间大致为5分钟到10分钟，12点到下午6点需要吃饭，耗时30分钟，则每班需要多少人，并给出巡检工人的巡检路线和时间表。

Considering that the inspectors need to rest every 2 hours, the time is roughly 5 minutes to 10 minutes, they need to eat from 12:00 to 6:00 pm, and it takes 30 minutes, how many people are needed for each shift, and give the inspection workers' patrol Check routes and timetables.

如果采用错时上班，出现计算前两问的结果，分析错时上班是否更节省人力。

Using staggered time to go to work, the results of calculating the first two questions appear, and analyze whether going to work at the wrong time saves manpower.

**模型准备：**

**Floyd算法求解最短距离矩阵**

**Floyd's algorithm to solve the shortest distance matrix**

在进行最佳巡检路线规划之前，需要知道各巡检节点之间的最短距离。因此，这里引入Floyd算法，求解26个巡检节点之间的最短距离矩阵。Floyd算法又称为插点法，是一种利用[动态规划](https://baike.baidu.com/item/%E5%8A%A8%E6%80%81%E8%A7%84%E5%88%92/529408" \t "_blank)的思想寻找给定的[加权图](https://baike.baidu.com/item/%E5%8A%A0%E6%9D%83%E5%9B%BE/10579361)中多源点之间[最短路径](https://baike.baidu.com/item/%E6%9C%80%E7%9F%AD%E8%B7%AF%E5%BE%84/6334920)的算法。

结合附件给出的巡检路线图，可以发现任意巡检节点$i$ 到$j$ 的最短路径有两种可能：

1. 直接从i到j；
2. 从i经过若干个节点k到j。

针对第二种情况，需要采用特殊处理，采用[松弛操作](https://baike.baidu.com/item/%E6%9D%BE%E5%BC%9B%E6%93%8D%E4%BD%9C)，对在i和j之间的所有其他点进行一次松弛，穷举连通$i,j$ 的可能路径，设路径经过的中间节点为${{k}\_{1}},{{k}\_{2}},\cdots ,{{k}\_{m}}$，算法中求解两节点之间最短距离的状态转移方程为：



其中$path[i,j]$ 表示从巡检点$i$ 到巡检点$j$ 之间的最短距离。初始化时，当$i,j$ 不相邻时，$path[i,j]=\infty $，$path[i,i]=0$

由此确定未直接联通的各巡检点之间的最短距离，从而得到所有节点之间的最短距离矩阵

方便后续确定最少工作人数和规划最佳巡检路线。

此算法简单有效，由于三重循环结构紧凑，对于稠密图，效率要高于[Dijkstra算法](https://baike.baidu.com/item/Dijkstra%E7%AE%97%E6%B3%95" \t "_blank)和[SPFA算法](https://baike.baidu.com/item/SPFA%E7%AE%97%E6%B3%95)。



****

****

****

****

**模型一：**

**模型构建**

**初始单人巡检路径设定**

。

初始单人巡检路径的设定也只是为了找出初始的最少巡检人数，所以初始巡检路径可以不是最优的。由于最优路径是嵌套在工作量均衡性和整体耗时最短的基础上的，如果直接用遗传算法求解出最短路径，求解复杂度高，运算速度慢**。所以本文引入最小生成树，将其作为初始巡检路径。**

最小生成树是用最小代价遍历整个图中所有顶点，所有的权值和最小，可以有效保证求解得到的初始巡检路线接近最短路径，且求解的时间空间复杂度较小，**可提高模型的整体求解效率。**。

克鲁斯尔算法主要是针对边来展开，边数少时效率会非常高，所以对于本文稀疏的巡检图有很大的优势，下面描述一下利用快速排序和并查集实现的Kruskal算法求解初始巡检路径（最小生成树）的步骤：

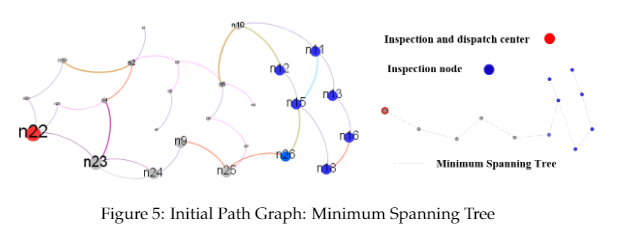
设工人需要巡检的路径网络是一个带权无向量图$G(V,E,W)$ ，其中$V$ 代表巡检员需要巡检的所有节点，$E$ 表示联通各巡检点的路径集，$W$ 表示各路径长度的集合。

1. 首先将所有路径$E$ 按照其长度的大小进行排序，设需要构造的最小生成树为$T$ 。
2. N个巡检节点看成独立的n棵树组成的森林；
3. 按长度从小到大枚举路径，如果所选的路径$e(u,v)$ 连接的巡检节点$u,v$ 属于两棵不同的树，则生成最小生成树的一条边，并将这颗树合并为一棵树。
4. 重复（3），直到选完$n-1$ 条边为止。

算法的实现需要一些额外的知识，首先是要从小到大枚举边，这里可以用快速排序直接将边集排序，或者可以用小顶堆来初始化边集。

然后是要判断两个点是否在统一集合中，还需要将两个集合合并到一个集合中。这里要用到并查集这一数据结构。这方面的知识请自行参考相关资料。

下面是利用快速排序和并查集实现的Kruskal算法。



如图5所示的求解到的巡检路径最小生成树，则巡检人员的初始巡检路径${{T}\_{path}}$ 为${{n}\_{22}}\Rightarrow {{n}\_{23}}\Rightarrow {{n}\_{24}}\Rightarrow {{n}\_{9}}\Rightarrow {{n}\_{25}}\Rightarrow {{n}\_{26}}\Rightarrow {{n}\_{15}}$ $\Rightarrow {{n}\_{12}}\Rightarrow {{n}\_{15}}\Rightarrow {{n}\_{18}}\Rightarrow {{n}\_{16}}\Rightarrow {{n}\_{13}}\Rightarrow {{n}\_{11}}$

首先由计算机随机分组，将26个巡检节点随机分配给n名巡检人员，再由Kruskal算法求解得到最小生成树，并沿着最小生成树的路线产生初始巡检路径，从而得到第 个工作人员的任务分配结果 ，

其中 表示第i个巡检人员的巡检路线；：分配的需要巡检的节点；

：巡检各节点的耗时； ：总路线上所有子边的集合 ； ：每条子路径步行时长的集合。

则第i个巡检工人的单次巡检总时长为：



则n个工人将26个巡检节点全部巡检完的总时长尽可能少的目标函数为



为让所有工作量尽可能均衡，同时也可以方便后续的排班，则每个巡检人员单次巡检的时长的方差应尽可能小，则得到工作量均衡的目标函数：



首先由计算机随机分组，将26个巡检节点随机分配给n名巡检人员，再由Kruskal算法求解得到最小生成树，并沿着最小生成树的路线产生初始巡检路径，从而得到第$i$ 个工作人员的任务分配结果：

\[T\_{path}^{i}({{V}^{i}},V{{T}^{i}},{{E}^{i}},{{W}^{i}})\] ，

其中$T\_{path}^{i}$ 表示第i个巡检人员的巡检路线；${{V}^{i}}=\left\{ v\_{1}^{i},v\_{2}^{i},\cdots ,v\_{k}^{i} \right\}$：；分配的需要巡检的节点

$V{{T}^{i}}=\left\{ vt\_{1}^{i},vt\_{2}^{i},\cdots ,vt\_{k}^{i} \right\}$：巡检各节点的耗时；${{E}^{i}}=\left\{ e\_{1}^{i},e\_{2}^{i},\cdots ,e\_{k-1}^{i} \right\}$ ：总路线上所有子边的集合 ；${{W}^{i}}=\left\{ w\_{1}^{i},w\_{2}^{i},\cdots ,w\_{k-1}^{i} \right\}$ ：每条子路径步行时长的集合。

则第i个巡检工人的单次巡检总时长为：

$Tim{{e}\_{i}}=\sum\limits\_{\tau =1}^{k}{vt\_{\tau }^{i}}+\sum\limits\_{\tau =1}^{k-1}{w\_{\tau }^{i}}$

则n个工人将26个巡检节点全部巡检完的总时长尽可能少的目标函数为

$\text{min }Tim{{e}\_{total}}=\sum\limits\_{i=1}^{n}{\left[ \sum\limits\_{\tau =1}^{k}{vt\_{\tau }^{i}}+\sum\limits\_{\tau =1}^{k-1}{w\_{\tau }^{i}} \right]}$

为让所有工作量尽可能均衡，同时也可以方便后续的排班，则每个巡检人员单次巡检的时长的方差应尽可能小，则得到工作量均衡的目标函数：

$\min \text{ }\frac{1}{n}{{\sum\limits\_{i=1}^{n}{\left[ Tim{{e}\_{i}}-E\left( \sum\limits\_{i=1}^{n}{Tim{{e}\_{i}}} \right) \right]}}^{2}}$

在节点巡检的总耗时为67分钟，各节点之间的路径总距离为78分钟，而70%的节点的巡检周期为35分钟，则最少的巡检人数应满足如下约束：

$\min \text{ num}\ge (67+78)/(35+\varepsilon )$

其中$\varepsilon $ 表示平均巡检周期的波动量，由于其中八个巡检节点的巡检周期大于35分钟，因此$\varepsilon $ 一定为正值。当$\varepsilon =0$ 时，算得$num\ge 4.14$ 。剔除最大值后，所有节点巡检周期平均值为66分钟，此时算得$num\ge 2.2$，说明最少的人数不会少于3.因此n从3开始，依次代入模型中，对每一个巡检人员的任务分配进行求解。

（只检验特定人数下，换班后的第一次排班是否满足工作要求）如果工厂采用固定时间上班，换班初期，巡检人员需要从22号巡检调度中心出发前往分配的巡检节点，这个时候路途较长，巡检人员最容易耽误巡检周期。所以考察巡检人数是否满足要求时，仅需检验刚换班后巡检员从巡检中心出发的初次巡检耗时是否超出各节点的巡检周期便可。

（指定一个人专门负责巡检八个巡检周期长的节点）针对巡检任务的分配，通过观察各节点巡检周期的大小，可以发现，只有八个节点的巡检周期大于35分钟，且需要的检查时间间隔较长，可以单独让一个巡检员完成这些节点的巡检。

（给巡检人员分配固定的工作节点，节点分布尽可能集中）但是除了刚开始时工作人员需要从调度中心出发之外，其他时候他们都只需要在自己负责的区域进行巡检。因此，应该让巡检人员的巡检节点尽可能集中，这个时候工作人员后续的巡检时长更少。

结合上述分析可以知道，给工人分配固定的巡检节点时，不需要考虑从巡检中心出发的情况。直接将26个节点随机分配给巡检人员，通过模拟退火算法，搜索让总工作时长和工作尽可能均衡的两个目标函数（3）（4）取得最优解的巡检策略。

（巡检人数判定的简便方法）根据模拟退火算法求解得到节点分配方案，计算从巡检中心出发，按照最小生成树产生的路径，完成所有节点的巡检的工作耗时。如果每个巡检人员完成任务的工作时长都小于35分钟，说明此时的巡检人数一定大于或等于最小巡检人数。因为所有巡检节点的巡检周期都大于等于35分钟，且最难完成任务的时间段是换班后的第一次排班，全部人员需要从巡检中心出发。

结合上述分析可以知道，给工人分配固定的巡检节点时，不需要考虑刚换班后从巡检中心出发的情况。模型的整体思想是，将26个节点随机分配给巡检人员，对分配的节点运用Kruskal算法，求解出连接各节点的初始巡检路径，并运算单个人和所有工人完成一次巡检所需要的工作时长，最后计算所有工作时长的方差，将其作为衡量各工人工作均衡性的指标。

再运用模拟退火算法，多次运算搜索，记录下每次迭代的结果与前一时刻比较，当达到终止条件时，输出结果，便得到特定巡检人数下的最优巡检策略。在算法迭代过程中，需使得单个工人的工作耗时尽可能少，所有工人的总巡检时长尽可能短，每个巡检工人的工作量尽可能均衡，可以大致看作如下三个目标函数的求解：

$\min \left\{ \begin{align}

& \text{ }Tim{{e}\_{i}}=\sum\limits\_{\tau =1}^{k}{vt\_{\tau }^{i}}+\sum\limits\_{\tau =1}^{k-1}{w\_{\tau }^{i}\text{ },\text{ Individual working hours}} \\

& \text{ }Tim{{e}\_{total}}=\sum\limits\_{i=1}^{n}{\left[ \sum\limits\_{\tau =1}^{k}{vt\_{\tau }^{i}}+\sum\limits\_{\tau =1}^{k-1}{w\_{\tau }^{i}} \right]\text{ },\text{ Total working hours}} \\

& \text{ }\frac{1}{n}{{\sum\limits\_{i=1}^{n}{\left[ Tim{{e}\_{i}}-E\left( \sum\limits\_{i=1}^{n}{Tim{{e}\_{i}}} \right) \right]}}^{2}}\text{ },\text{ Workload variance} \\

\end{align} \right.$

