Day4 最短路

A

dijkstra模板。

B

floyd模板题

C

SPFA模板，或者使用dijkstra的优先队列优化模板。

D

建图跑dijkstra。注意dis数组的初始值需要至少10^{18}。记录路径可以再松弛操作的时候，记录某一点的前继节点，然后递归的输出即可。如：

```
void dijkstra() {
    ......
    for (int i = head[u]; ~i; i = e[i].next) {
        ......
            if (dis[v] > dis[u] + w) {
                ......
                pre[v] = u;
            }
    }
    ......
}
void print(int x) {
    if (x == -1) {
        return;
    }
    print(pre[x]);
    cout << x << " ";
}
```

E

建立一张负边权的图。使用SPFA判断是否存在负环。如果存在则为YES。

F

题意：给出n个车站的坐标，和每一单位距离的花费d，从x到y的花费定义为$dis(x,y)*d$。每到达一个车站i都会免去$a[i]$的花费，问从1到n的最少花费是多少。

相当于给定了点，经历一张完全图，每个边的距离分别是$(|x_i-x_j|+|y_i-y_j|)*d-a_i$

由于点数很少，所以可以直接跑Flody，写的快。

G

相遇的情况，可以分为在点相遇，或者在边上相遇，对其分类讨论。

首先用Dijkstra算法，记录S到T的最短路径长度为L，路径数量为cnt。根据乘法原理，全部的方案数量为$f=cnt^2$。

然后再次跑最短路，记录从 S 出发走到点x的路程为 d_1，方案为f_1，记录从 T 出发走到点x的路程为d_2，方案为f_2。

如果考虑在点x相遇的情况，当且仅当$d_1=d_2=\frac{L}{2}$时候相遇，其方案数量为$g={f_{x1}}^2*{f_{x2}}^2$。

如果考虑在边(u,v)相遇的情况，当且仅当$2*dis_u<L\land 2*dis_v<L$时候相遇，其方案数量为$z={f_{u1}}^2*{f_{v2}}^2$。

所以最后的方案数为：$f-g-z$，注意取模的情况。

时间复杂度：$O(nlog(n+m))$