航班安排

【背景】

1.wqs 爱好模拟飞行。

2.clj 开了一家神犇航空，由于 clj 还要玩游戏，所以公司的事务由你来打理。

注意：题目中只是用了这样一个背景，并不与真实/模拟飞行相符

【题目描述】

神犇航空有 K 架飞机，为了简化问题，我们认为每架飞机都是相同的。神犇航空的世界中有 N 个机场，以 0..N-1 编号，其中 0 号为基地机场，每天 0 时刻起飞机才可以从该机场起飞，并不晚于 T 时刻回到该机场。一天，神犇航空接到了 M 个包机请求，每个请求为在 s 时刻从 a 机场起飞，在恰好 t 时刻到达 b 机场，可以净获利 c。设计一种方案，使得总收益最大。

【输入格式】

第一行，4 个正整数 N,M,K,T，如题目描述中所述；

以下 N 行，每行 N 个整数，描述一个 N*N 的矩阵 t，$t_{i,j}$ 表示从机场 i 空载飞至机场 j，需要时间 $t_{i,j}$；

以下 N 行，每行 N 个整数，描述一个 N*N 的矩阵 f，$f_{i,j}$ 表示从机场 i 空载飞至机场 j，需要费用 $f_{i,j}$；

以下 M 行，每行 5 个整数描述一个请求，依次为 a,b,s,t,c。

【输出格式】

仅一行，一个整数，表示最大收益。

【样例输入】

2 1 1 10

0 5

5 0

0 5

5 0

0 1 0 5 10

【样例输出】

5

【数据规模及约定】

对于 10%的测试数据，K=1；

另有 20%的测试数据，K=2；

对于全部的测试数据，N,M<=200，K<=10，T<=3000，$t_{i,j}$<=200，$f_{i,j}$<=2000，0<=a,b<N，0<=s<=t<=T，0<=c<=10000，$t_{i,i}=f_{i,i}=0$，$t_{i,j}<=t_{i,k}+t_{k,j}$，$f_{i,j}<=f_{i,k}+f_{k,j}$。

【参考算法一】

对于 10%的测试数据，K=1。可以使用动态规划求解，F[i][j]表示飞机在 i 时刻 j 地时所能获得的最大收益。

时间复杂度：O(NT)

期望得分：10 分

【参考算法二】

另有 20%的测试数据，K=2。由于每架飞机的行程为 0->某个请求的起点->(执行请求)->该请求的终点->另一请求的起点->…->0，所以动态规划只需记录两架飞机最后执行的请求即可。F[i][j]表示两架飞机分别执行完 i,j 请求，转移为 F[i][j]=max(F[i][j], F[j][i])，及 F[k][j] = max(F[k][j], F[i][j] - f[b[i]][a[k]] + c[k])，s[a[k]] - t[b[i]] ≥ t[b[i]][a[k]]。状态数 $O(N^2)$，转移数

O(N)(认为 N,M 同数量级)。

时间复杂度：$O(N^3)$

期望得分：**20** 分

30 分(结合算法一)

【参考算法三】

此问题可以用最大费用流求解，流量代表飞机的移动，费用代表收益(可能为负)，源点代表 0 时刻所有飞机都在 0 号机场，汇点代表 T 时刻所有飞机都回到 0 号机场。每个请求建成起终两个点，连一条边代表执行这个请求。考察每对请求的终点和起点(源点和汇点也可以分别认为是某个想象的请求的终点和起点)，如果有足够的时间从一个的终点到达另一个的起点，那么连一条边代表执行完前一个请求后飞往后一个的起点准备执行后一个请求，这样建图后求出源点到汇点限制流量不超过 K 的最大费用流即为答案。

时间复杂度：$O(max\text{-}fee\text{-}flow(N,N^2))$

期望得分：**100** 分

【费用流的优化】

这里有一个优化是每次增广之后可以不用 SPFA 重新求最长路，可以仅在第一次增广前用一次 SPFA，每次增广后可以用 Dijkstra 算法更新最长路。若记增广前的最长路值为 d，增广后重新计算过程中的最长路值为 d’(计算完成后即为更新后的最长路)，则 Dijkstra 中每次找出 d’-d 最大且未扩展的点进行扩展，这种做法等价于把每条边的权 w(i,j) 替换为 w(i,j)+d[i]-d[j]后求最长路。若增广后的残余网络中存在一条边(i,j)，那么增广前要么同样存在边(i,j)，要么存在其反向边(j,i)，无论哪种情况，都可以保证 w(i,j)+d[i]-d[j]<0，所以这种做法是正确的。这样可以以更低的复杂度求出答案。