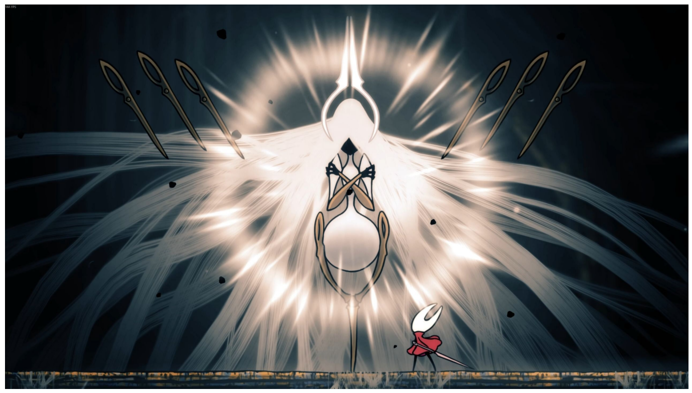
F. 丝之歌

时间限制：2.0 s　内存限制：512 MB



小 C 是一位幸运的《空洞骑士：丝之歌》玩家。在游戏中，玩家扮演致命猎手“大黄蜂”，踏入丝与歌统治的王国。被俘至陌生世界，在致命的朝圣之路上迎战强敌、破解远古谜团，直抵王国巅峰。作为一位老空洞玩家，小 C 自从丝之歌发售以来就没日没夜地乐在其中。但不知道是因为手感生疏还是这一作难度加强，他发现自己在这一作中经常死亡，于是小 C 决定对游戏难度进行深入研究。

游戏中核心的货币系统以”念珠 “为单位，绝大部份商品都需要使用念珠购买。但遗憾的是，玩家死亡时全部的念珠会丢失，无法找回。但游戏友善地提供了“念珠串”机制——玩家可以在**商人处**把若干念珠兑换成念珠串，念珠串在玩家死亡时不会消失，在玩家重生后依然存在。具体的，如果玩家当前拥有的念珠不少于 a 个，那么他可以消耗 a 个念珠兑换一个念珠串。在一个商人处玩家可以兑换任意数量的念珠串，只要符合兑换的要求。此外，在任意时刻玩家都可以选择拆解念珠串。每拆解一个念珠串可以使得当前的念珠数量增加 b 个 $(b \le a)$，同时该念珠串消失。

游戏中玩家的生命值以“面具”为单位。在满血状态下玩家拥有 c 个面具，当玩家的面具数变为 0 时玩家死亡，会直接回到最近的复活点复活，并回到满血状态。游戏中只有复活点处可以回复血量。游戏中共有 m 种不同类型的敌人。在面对类型为 i 的敌人时，小 C 对于战斗的结果用一个下三角矩阵 P_i 刻画：$P_{i,x,y}$ $(1 \le x \le c, 0 \le y \le x)$ 代表小 C 在与一个类型为 i 的敌人战斗前有 x 个面具，战斗后有 y 个面具的概率。特殊的，$y = 0$ 代表小 C 战斗后死亡。如果小 C 没有战斗后死亡，则代表小 C 击杀了该敌人，将会获得 w_i 个念珠。

为了简化模型，小 C 把游戏抽象成线性设计：游戏共有 n 个关卡，他需要按照关卡编号从小到大依次通过。每个关卡由三部分构成：复活点、敌人、商人。在关卡开始时玩家会先经过复活点，并把血量回复至满血状态。然后玩家要**依次**面对若干个敌人。如果小 C 在挑战某个敌人时死亡，则会按照上述规则回到该关卡的复活点，再重新一次挑战所有敌人（该关卡内此前击败的敌人会重置回初始状态）。如果小 C 顺利击杀该关卡内**全部敌人**，则可以顺利来到商人处合成念珠串或购买商品，并进入下一关。

小 C 知道自己现在的游戏水平不佳，因此**除了第 n 个关卡的商人处**，每次他来到商人处都会尽可能的把手头的念珠兑换成念珠串，直到手头的念珠数量小于 a 。

小 C 认为来到第 n 个关卡的商人处视为游戏通关，并且他会在第 n 个关卡的商人处拆解此前兑换的全部念珠串。他想知道在这样的模型假设下，他通关时拥有的念珠期望个数是多少？答案对 998244353 取模。

注：题面中描述的若干机制和现实中该游戏机制略有出入，请以题面描述为准。

Input

第一行包含五个整数 n, m, a, b, c $(1 \le n, m, a, b \le 10^3, 1 \le c \le 10)$，代表游戏的关卡数，敌人的种类数，兑换一个念珠串所需的念珠数，拆解一个念珠串可以获得的念珠数，满血状态下玩家的面具数。

第二行 m 个整数 $w_1, w_2, \cdots, w_i$ $(1 \le w \le 10^6)$，其中 w_i 代表击杀第 i 类敌人能获得的念珠数量。

接下来输入 m 个矩阵，每个矩阵占用 c 行，其中第 x 行有 $x+1$ 个整数。其中第 i 个矩阵的第 x 行的第 y 个数字为 $p_{i,x,y-1}$ $(1 \le p_{i,x,y-1} \le 10^6)$。令 $sum_{i,x} = \sum_{0 \le y \le x} p_{i,x,y}$，则对应第 i 个敌人的概率下三角矩阵计算方法为 $P_{i,x,y} = \frac{p_{i,x,y}}{sum_{i,x}}$ $(1 \le x \le c, 0 \le y \le x)$。

接下来 n 行，每行描述一个关卡的信息：

- 每行开头为一个整数 t_i $(1 \le t_i \le 100)$，代表这一个关卡中共有多少波敌人。
- 接下来 $2t_i$ 个整数 $ty_{i,1}, num_{i,1}, ty_{i,2}, num_{i,2}, \cdots, ty_{i,t_i}, num_{i,t_i}$ $(1 \le ty_{i,j} \le m, 1 \le num_{i,j} \le 10^9)$，代表第 i 个关卡的敌人共有 $\sum_{1 \le j \le t_i} num_{i,j}$ 个，依次为：$num_{i,1}$ 个种类为 $ty_{i,1}$ 的敌人，$num_{i,2}$ 个种类为 $ty_{i,2}$ 的敌人，以此类推。注意玩家会依次挑战这些敌人，每次只会面对一个敌人。

Output

输出一行一个整数，代表小 C 通关时期望拥有的念珠个数。

Sample Input 1	Sample Output 1
3 2 5 3 1 3 2 1 2 1 1 2 1 1 2 1 1 2 3 1 1 3	850356316

Hint

样例中共有 3 个关卡，2 类敌人，每凑够 5 个念珠即可兑换一个念珠串，每拆解一个念珠串可兑换 3 个念珠。玩家满血状态下只有 1 个面具。

对于第一类敌人，玩家有 $1/3$ 的概率被击败（变成 0 个面具），有 $2/3$ 的概率获胜。战胜一个第一类敌人可以获得 3 个念珠。

对于第二类敌人，玩家有 $1/2$ 的概率被击败，有 $1/2$ 的概率获胜。战胜一个第二类敌人可以获得 2 个念珠。

第一关依次是一个第一类敌人，一个第二类敌人，小 C 通关后总共能收获 3+2=5 个念珠。由于 5 个念珠正好足够兑换一个念珠串，小 C 会在第一关结束后兑换一个念珠串，然后手中不剩下任何念珠。第二关由三个第二类敌人构成，小 C 通关后总共能收获 2+2+2=6 个念珠。小 C 会在第二关结束后兑换一个念珠串，然后手中剩下 1 个念珠。

第三关由三个第一类敌人构成。如果小 C 第一次尝试第三关就通关，那么手中的念珠数为（此前剩下的）1+3+3+3=10 个念珠，对应的概率为 $(\frac{2}{3})^3 = \frac{8}{27}$。否则小 C 在第三关死亡时会遗失手中的所有念珠，经历若干次重生后通关第三关时，手中的念珠数为 3+3+3=9 个念珠，对应的概率为 $1 - \frac{8}{27} = \frac{19}{27}$。

加上此前积累的两个念珠串，小 C 最终手中的念珠数量期望数为：

$3 + 3 + (10 \times \frac{8}{27} + 9 \times \frac{19}{27}) = 15 + \frac{8}{27} \equiv 850356316 \pmod{998244353}$