

D. Regularity Race

Time limit: 1s

Um tipo de corrida de automóveis, veículos em geral, mesmo bicicleta ou a pé, é a corrida de regularidade. Neste tipo de corrida existe um percurso a ser percorrido e uma velocidade média a ser mantida. Para garantia existem vários pontos de controle de tempo. Estes pontos são desconhecidos dos competidores para evitar que eles tentem corrigir o tempo na chegada ao ponto. A contagem em cada ponto de controle é de penalização por atraso ou adiantamento em pontos. Uma corrida de Enduro a Pé, possui vários pontos de controle. Em cada ponto de controle, se o corredor (ou equipe) atrasar, é punido com 1 ponto por segundo de atraso, com limite de 1200 pontos por ponto de controle (20 minutos de atraso) ou 2 pontos por segundo de adiantamento, com limite de 1200 pontos por ponto de controle (10 minutos). Vence a prova quem tiver menos pontos. Como critério de desempate a equipe inscrita de menor número.

Entrada

A entrada contém vários casos de teste. Cada caso de teste utiliza várias linhas da entrada, começa com uma linha indicando três números inteiros: **V**, **P**, **T**: $0 < V < 20$, a velocidade média, em km/h, a ser mantida na corrida; $0 < P < 10$ o número de pontos de controle; e $0 < T < 20$ o número de times competidores. Na segunda linha estão **P** números inteiros indicando a posição de cada ponto de controle, em metros, a partir da origem, $0 < P_i < 4.000$. O último ponto de controle coincide com o final da corrida. As próximas **T** linhas representam **P** valores inteiros indicando os tempos de cada competidor, em segundos, em cada ponto de controle, time 1 na linha 1, e assim sucessivamente. Os casos de entrada terminam com **V = P = T = 0**.

Saída

Para cada caso de teste da entrada seu programa deve gerar, na saída, uma linha única com a classificação final da corrida, listando os times pelos seus números com um único espaço em branco entre cada time.

Exemplo de Entrada	Exemplo de Saída
10 5 4	4 3 2 1
3000 5000 9500 12000 14500	2 3 1
1090 1820 3460 4400 5350	
1070 1790 3410 4310 5210	
1070 1810 3420 4310 5240	
1100 1820 3420 4320 5210	
10 3 3	
3000 6000 10000	
1070 2160 3610	
1080 2170 3610	
1070 2160 3600	
0 0 0	

Este problema fez parte da CPU2016 (Competição de Programação Universitária) da Universidade Estadual de Santa Cruz - UESC.

Por Hamilton José Brumatto, UESC  Brazil

