

Gerenciamento de Partição Fixa em Competição

Contest Local, Universidade de Ulm  Alemanha

Timelimit: 1

Uma técnica usada em estratégias iniciais de competição de programação envolve particionar a capacidade intelectual disponível de uma equipe por um número de membros, com cada membro tendo uma quantia fixa de inteligência, e diferentes membros potencialmente tendo quantidades diferentes. A soma da capacidade de todos os membros é igual a capacidade intelectual total da equipe.

Dado um conjunto de problemas, a tarefa da equipe é atribuir os problemas a diferentes membros da equipe, de modo que os problemas possam ser resolvidos simultaneamente. Esta tarefa é difícil devido ao fato de que o tempo para a solução de um problema pode depender da quantidade de informações disponíveis. Todo problema tem um requisito mínimo de inteligência: se a atribuição for para um membro mais brilhante, o tempo de solução pode aumentar ou diminuir.

Nesta tarefa, você tem que determinar atribuições ideais de problemas aos membros da equipe. O programa informa as capacidades intelectuais dos membros da equipe disponíveis para a solução dos problemas, e uma descrição, para cada problema, de como o tempo de solução depende da quantidade de informação disponível. O programa tem que encontrar um cronograma de solução que minimiza o tempo médio de solução para os problemas. Um cronograma de solução é uma atribuição de problemas aos membros da equipe em relação ao tempo, de tal forma que não há dois problemas que usam o mesmo membro ao mesmo tempo, e não há problema que é atribuído a um membro da equipe com menos capacidade do que o seu requisito mínimo. O tempo para a solução do problema é a diferença entre o tempo em que o problema foi submetido para ser resolvido (o início da competição começa no tempo zero para todos os problemas desta tarefa), e o tempo em que o problema foi resolvido.

Entrada

Os dados de entrada conterão vários casos de teste. Cada caso de teste começa com uma linha contendo um par de números inteiros m e n . O número m especifica o número de membros da equipe ($1 \leq m \leq 3$), e n especifica o número de problemas a serem resolvidos ($1 \leq n \leq 10$).

A próxima linha contém m inteiros positivos informando os valores de inteligência dos membros da equipe m . Em seguida, as n linhas descrevem as compensações de tempo de capacidade para cada um dos n problemas. Cada linha começa com um k inteiro positivo ($k \leq 10$), seguido por pares de k inteiros positivos $s_1, t_1, s_2, t_2, \dots, s_k, t_k$ que satisfazem $s_i < s_{i+1}$ para $1 \leq i < k$. O requisito mínimo de inteligência do problema é s_1 , ou seja, não pode ser resolvido por um membro com menos capacidade intelectual do que este número. Se o problema é resolvido por um membro da equipe com capacidade s , onde $s_1 \leq s < s_{i+1}$ para algum i , então o tempo de solução será t_i . Finalmente, se o problema é resolvido por um membro da equipe com capacidade intelectual s_k ou mais, então o tempo de execução será t_k .

Um par de zeros será a entrada para o último caso de teste.

Você pode assumir que cada problema será resolvido exatamente no tempo especificado para a capacidade informada, independentemente do número de outros problemas que estão sendo resolvidos por outros membros da equipe ao mesmo tempo. Nenhum problema terá um requisito de inteligência maior do que a do membro da equipe mais brilhante.

Saída

Para cada caso de teste, primeiro apresentar o número do caso (começando com 1 e aumentando sequencialmente). Em seguida, imprima o tempo médio de solução para o conjunto de problemas com dois

dígitos à direita do ponto decimal. Siga a descrição de um cronograma de solução que alcança esse tempo médio de solução. Visualiza uma linha para cada problema, na ordem em que foi informado na entrada, que identifica o número do problema, o membro utilizado para resolver (numerado de acordo com a ordem de entrada), o tempo que o membro começou a resolver o problema, e o tempo em que o problema foi resolvido. Siga o formato mostrado no exemplo de saída, e imprima uma linha em branco após cada caso de teste.

Exemplo de Entrada	Exemplo de Saída
2 4 40 60 1 35 4 1 20 3 1 40 10 1 60 7 3 5 10 20 30 2 10 50 12 30 2 10 100 20 25 1 25 19 1 19 41 2 10 18 30 42 0 0	Case 1 Average solution time = 7.75 Problem 1 is solved by member 2 from 0 to 4 Problem 2 is solved by member 1 from 0 to 3 Problem 3 is solved by member 1 from 3 to 13 Problem 4 is solved by member 2 from 4 to 11 Case 2 Average solution time = 35.40 Problem 1 is solved by member 3 from 19 to 49 Problem 2 is solved by member 2 from 0 to 25 Problem 3 is solved by member 3 from 0 to 19 Problem 4 is solved by member 2 from 25 to 66 Problem 5 is solved by member 1 from 0 to 18