Анализ на задача **Expedition**

1. **Подзадача 1 (11 точки) - пълно изчерпване**

За решаването на тази подзадача е достатъчно да проверим всички възможности за избиране на автобус за всеки студент и след това да изчислим необходимата сума за това разпределение. Сложността на алгоритъма е *O(M^N)*.

2. **Подзадача 2 (26 точки) - всеки студент заплаща за наемане на автобус.**

Забелязваме, че ако един от автобусите е "по-изгоден" от друг за даден студент на позиция X, то за всички студенти на позиции Y, вдясно от X, това свойство ще се запази, тъй като цената за прибиране на студент Y, ако използва автобус B, ще бъде с v * (Y-X) (не зависи от B) повече спрямо цената за прибирането на X, използвайки автобус B. Така можем да сортираме автобусите и студентите по координати заедно. Когато ги обхождаме ще поддържаме най-изгодния до момента автобус B и ако се намираме на автобус, който е по-изгоден да обновим B. Ако пък се намираме на студент, то трябва просто да пресметнем цената за стигането му до автобус B + наема за автобуса.

3. **Подзадача 3 (28 точки) - оптимизиране с битови маски**

За решаването на тази задача нека с $f_{(mask, i)}$ означаваме минималната сума, която трябва да се заплати, за да приберем първите i студенти, като сме наели автобус номер j ако (j-1)-ия бит в двоичнота представяне на mask е единица. Иначе казано в mask ни е записана информацията за наетите автобуси. Тогава за $f_{(mask, i)}$ имаме:

$f_{(mask,\ i)} = \infty$, ако за всеки автобус j, който сме взели в hash е изпълнено $x_i < y_j$

Иначе $f_{(mask, i)} = \min (f_{(mask, i-1)} + (x_i-y_j)*v_i$, за всеки автобус j, който сме взели в mask).

В началото трябва да пресметнем $f_{(mask, 0)} = c_{j1} + c_{j2} + \ldots + c_{jp}$, като това са наемите за всички автобуси, които сме наели в mask.

<u>Наблюдение: Нека знаем, че при оптималното разпределение на всичките M студенти, студент i отива до автобус номер j. Тогава студент i+1 не би било оптимално да отиде до автобус с номер, по-малък от j. Така забелязваме, че до кой автобус ще отиде студент i+1 зависи единствено от автобуса, при който отива студент i, а не от предходните.</u>

4. **Подзадача 4 (51 точки)**

Така идва идеята за прилагане на динамично оптимиране: нека с $dp_{(i,j)}$ означим минималната цена, която трябва да се заплати, за да се приберат първите i студенти, като i-тия се качва на j-тия автобус. Използвайки наблюдението по-горе получаваме следната формула:

$dp_{(i,j)} = \infty$, ако е изпълнено $x_i < y_j$, иначе:

$dp_{(i,j)} = (x_i-y_j)*v_i + \min (dp_{(i-1,j)}, dp_{(i-1,j-1)} + c_j, dp_{(i-1,j-2)} + c_{j, \dots,} dp_{(i-1,1)} + c_j)$

Сложността на решението е *O(N)* за всеки стейт и общо *O(MN²)* За целия алгоритъм.

5. Подзадача 5 (74 точки)

За решаването и на последната подзадача се нуждаем от малка оптимизация на гореспоменатото изчисляване на $dp_{(i,j)}$. Забелязваме, че формулата се състои условно от два случая - случаят, в който i-тият студент отива при същия автобус като (i-1)-вия, където не добавяме отново цената за наемане на j-тия автобус и случая, в който i-тия отива при различен автобус и заплаща цената за наемането му. Именно втория случай изчисляваме със сложност O(N) за всеки стейт и можем да го оптимизираме. Нека с $l_{(i,j)}$ означаваме $\min(dp_{(i,1),} dp_{(i,2)} \dots , dp_{(i,j)})$. И получаваме следната зависимост:

$dp_{(i,j)} = \infty$, ако е изпълнено $x_i < y_j$, иначе:

$dp_{(i,j)} = (x_i-y_j)*v_i + \min (dp_{(i-1,j)}, l_{(i-1,j-1)} + c_j)$

$l_{(i,j)} = \min(l_{(i,j-1)}, dp_{(i,j)})$

Сложността за изчисляване е *O(1)* за всеки стейт и *O(NM)* за цялото решение. Отговорът накрая се получава в $l_{(1,N)}$, $l_{(2,N)}$ … $l_{(M,N)}$. Така реализирано, решението би използвало прекалено много памет, но забелязваме, че можем да приложим стандартна оптимизация за намаляване на необходимата памет, като пазим само стойностите от предходния ред, за да изчислим текущия.

Алтернативно решение би било за всеки студент i да фиксираме автобус j и да изчислим сумата за това разпределение, като всички студенти между автобуса и i-тия студент ще отидат при автобус j, а за останалите вече сме пресметнали отговора.

Автори: Илиян Йорданов,
Румен Михов,
Кинка Кирилова-Лупанова