

Task Expedition analysis

1. Subtask 1 (11 points) - brute force

To solve this part of the problem, it is sufficient to check all bus selection possibilities for each student and then calculate the required amount for this distribution. The complexity of the algorithm is $O(M^N)$.

2. Subtask 2 (26 точки) - Every student has to pay for renting a bus even if several students travel together

Note that if one of the buses is "more favorable" than another for a given student at position X , then for all students at positions Y to the right of X , this property will hold, since the price for the return of student Y if uses bus B , will be $v * (Y-X)$ (doesn't depend on B) more than the cost for the return of X using bus B . So we can sort buses and students by coordinates together. When we go through them, we will keep the most favorable bus B so far, and if we are on a bus that is more favorable, we will update B . If we are at a student, then we just need to calculate the cost of going to bus B + the rent for the bus B .

3. Subtask 3 (28 points) - optimization using bit masks

To solve this problem, let's $f_{(mask, i)}$ denote the minimum amount that must be paid for the return of the first i students, having hired bus number j if the $(j-1)^{th}$ bit in the binary representation of $mask$ is 1. In other words, the information about the rented buses is recorded in our mask. Then for $f_{(mask, i)}$ we have:

$f_{(mask, i)} = \infty$ if for each bus j that we have taken in $mask$ $x_i < y_j$.

Else $f_{(mask, i)} = \min (f_{(mask, i-1)} + (x_i - y_j) * v_i, \text{ for every bus } j \text{ we have taken in } mask)$.

At the beginning, we need to calculate $f_{(mask, 0)} = c_{j1} + c_{j2} + \dots + c_{jp}$, which are the rents for all the buses we rented in $mask$.

Observation: Let us know that in the optimal distribution of all M students, student i goes to bus number j . Then it would not be optimal for student $i+1$ to go to a bus with a number less than j . Thus, we notice that which bus student $i+1$ will go to, depends only on the bus that student i goes to, not on the previous ones.

4. Subtask 4 (51 points)

Thus comes the idea of applying dynamic optimization: let $dp_{(i,j)}$ denote the minimum price that must be paid to get the first i students to Shumen and the i^{th} student uses the j^{th} bus. Using the observation above we obtain the following formula:

$$dp_{(i,j)} = \infty, \text{ if } x_i < y_j, \text{ else:}$$

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

The solution complexity is $O(N)$ for each state and a total of $O(MN^2)$ for the entire algorithm.

5. Subtask 5 (74 points)

To solve the last part of this problem, we need a small optimization of the above calculation for $dp_{(i,j)}$. We notice that the formula consists of two cases - the case in which the i^{th} student goes to the same bus as the $(i-1)^{\text{th}}$, where we do not pay again the rental price of the j^{th} bus, and the case in which i^{th} one goes to a different bus and pays the price for renting it. For each state it is the second case that we calculate with complexity $O(N)$ and that can be optimized easily. Let $l_{(i,j)}$ denote $\min(dp_{(i,1)}, dp_{(i,2)} \dots, dp_{(i,j)})$. We get the following dependency:

$$dp_{(i,j)} = \infty, \text{ if } x_i < y_j, \text{ else:}$$

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

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

The solution complexity is $O(1)$ for each state and $O(NM)$ for the entire solution. The answer is finally obtained in $l_{(1,N)}, l_{(2,N)} \dots l_{(M,N)}$. Implemented this way, the solution would use too much memory, but we notice that we can apply a standard optimization to reduce the memory required by keeping only the values from the previous row to calculate the current one.

An alternative solution would be for each student i to fix bus j and calculate the sum for this distribution, with all students between the bus and the i^{th} student going to bus j , and for the rest we have already calculated the answer.

*Authors: Iliyan Yordanov,
Rumen Mihov,
Kinka Kirilova-Lupanova*