The Bus Driver Problem

Uva11389

Time: 1 seconds

Problem Descriptions(1/2)

- In a city there are n bus drivers. Also there are n morning bus routes & n afternoon bus routes with various lengths.
- Each driver is assigned one morning route & one evening route.
- For any driver, if his total route length for a day exceeds d, he has to be paid overtime for every hour after the first d hours at a flat r taka / hour.

Problem Descriptions(2/2)

- **Your task is to assign one morning route & one evening route to each bus driver so that the total overtime amount that the authority has to pay is minimized.**

Input

- The first line of each test case has three integers n, d and r, as described above.
  - n: n drivers and n routings.
  - d: total length cannot be exceeded.
  - r: extra paid r / hour while exceeding length d.
- In the second line, there are n space separated integers which are the lengths of the morning routes given in meters.
- Similarly the third line has n space separated integers denoting the evening route lengths.
- The lengths are positive integers less than or equal to 10000.
- The end of input is denoted by a case with three 0 s.

Output

- For each test case, print the minimum possible overtime amount that the authority must pay.

Sample Input / Output

Input

```
2 20 5
10 15
10 15
2 20 5
10 10
10 10
0 0 0
```

End of input

Output

```
50
0
```

Brute Force

1. n afternoon routings
2. n evening routings
3. Select minimum amount of exceeding k hours from possibilities
4. O()

Greedy

1. Sorting n afternoon routings
2. Sorting n evening routings
3. Maximum afternoon + Minimum evening routing
4. O(nlogn+n)
5. O(nlogn)

```cpp
#include <iostream>
#include <algorithm>

int main()
{
    int n, d, r;
    int morning_route[101], evening_route[101];

    while ((std::cin>>n>>d>>r)&&(n || d || r))
    {
        for (int i = 0; i < n; i++)
            std::cin>>morning_route[i];
        for (int j = 0; j < n; j++)
            std::cin>>evening_route[j];

        std::sort(morning_route, morning_route + n);
        std::sort(evening_route, evening_route + n);

        int extra_pay = 0;

        for (int k = 0; k < n; k++)
        {
            if (morning_route[k] + evening_route[n-k-1] > d)
                extra_pay += ((morning_route[k] + evening_route[n-k-1] - d) * r);
        }
        std::cout<<extra_pay<<std::endl;
    }
}
```