Competitive Programming [CC 3032]

Tomás Moreira Ferreira – 202006594

Problem: Hockey Road Trip

The training levels on the roller hockey youth squads of the União Desportiva de Oliveirense are quite ambitious and, therefore, the kids are always looking for the best and most diverse teams to train and compete with to have the best conditions to compete (and try to win) the national finals at the end of the season.

This team is located further to the north of Portugal, in the district of Aveiro, and therefore already has experience playing against teams from northern Portuguese territory. Since then, it was agreed with the parents of the young athletes a weekend for the team to embark on an adventure through the clubs in the central area of Portugal, forming a true roller hockey tour through the clubs in that area who otherwise would only have the chance to play against in the national finals.

The team was so excited about this idea that they left on a trip without having a clear path to follow. As the usual guidance applications have a limited number of stops while calculating the routes, the kids quickly went around on social media to contact someone who could help them calculate the best route, having heard about your skills in programming.

You will be given the number of games that can be played (only one game per team), all the clubs through which Oliveirense can pass by, each one identified by 3 capital alphabet letters ([A-Z][A-Z][A-Z]), the respective index in the competitiveness ranking (from 1 to 5; the bigger the better), and how long, in minutes, it takes to travel several roads that connect pavilions of two different clubs. Your objective will thus be to **outline the most efficient route for the trip**, always prioritizing the quality of the opponents to be faced (competitiveness ranking) but also looking at the time the route takes: in case the route with higher total competitiveness has an higher time distance as well, there could be a situation where the lower time distance will be preferred, with a limit of one-fifth of the total route distance per point(20%). For example, if path A takes 200 minutes and has a total competitiveness index of 14 and path B takes 150 minutes but has a total competitiveness index of 13, then path B will be preferred since 13=14-i and 150<200-i*k <==> 150<200-40, with k=200/5=40 and i=(200-150)//k=1.

In case of equal total competitiveness valuations and equal time distance, will be preferred the path with the higher amount of teams with highly competitive index. For example, if paths A and B both have the same distance and 12 as the total competitive index amount, but path A passes by 2 clubs of index 5 and one club of index 2, while path B passes by 3 clubs of index 4, it will be preferred the path A since it passes for more higher-indexed clubs than B (in this case, 5-indexed).

It is guaranteed that the team will always play the maximum number of games expected and that this number will never exceed the number of pavilions to be considered. No team can be played against more than once and, since some of the kids are visiting the centre for the first time in their lives, no road can be repeated at the same route, thus being assured that every club pavilion has always two roads connecting it.

Sample Input

The first line of input will contain three integers: the clubs that can be accessed during the road trip (**C**), all the distinct roads connecting the pavilions of two clubs(**R**), and the number of games expected to be played (**xG**).

Then will follow **C** lines, each one containing a sequence of 3 capital letters representing distinct clubs and an integer **N** revealing the competitiveness ranking of that club.

After that, there will be **R** lines, each one referring to a route between two clubs (3-letter sequences separated by a space) and the time that it takes to drive by it (**dT**).

Sample Output

The output will be composed of two lines:

1. The first line is the single best route to take, while always starting at the UDO pavilion. The stops are delimited by spaces and hyphens.
2. The second and last line outputs the phrase "**Time:** " plus the total time the road trip takes.

Constraints

- **2 <= C <= 75**
- **C < R <= 2850**
- **1 <= G <= C**
- **5 < dT <= 500**

Sample Input 1

```
8 11 4
SCT 3
FIS 1
PFC 2
HCT 4
SCP 5
SIN 2
SLB 5
OEI 3
UDO FIS 135
UDO SCT 130
SCT PFC 45
SCT SIN 45
FIS HCT 60
FIS SLB 50
PFC SCP 35
SCP HCT 80
SCP OEI 30
SIN OEI 30
OEI SLB 25
```

Sample Output 1

```
UDO – FIS – SLB – OEI - SCP
Total Time: 240
```

Sample Input 2

```
2 3 2

FIS 1

SCT 3

UDO FIS 130

UDO SCT 135

FIS SCT 5
```

Sample Output 2

```
UDO - FIS - SCT

Total Time: 135
```