Dungeon Explorer II

Time Limit: 1 s
Memory Limit: 256 MB

Description

Same with **Dungeon Explorer,** Given **n** numbers of people, each with **name** and **strength** level. After input for each people, there will be also given **m** line between each person, each with **index1** and **index2** (meaning that person with **index1** can summon **index2** for team up but **not vice versa**). Given **indexTarget**, calculate the total strength level of the person team up with other person who has connection,

Input

- The first line contains an integer **n**, the number of people
- The next **n** lines contain pairs of **name** (string) and **strength** (int) level of each person
  The Next line contains an integer **m**, the number of connection
- The next **m** lines contain pairs of **index1** (int) and **index2** (int). Indicating that person with **index1** has connection with person with **index2** (Not Vice Versa)
- Last line is **indexTarget**, that you have to calculate the total strength

Output

- Just print the Total strength for **name** teaming up with other connections

Constraints

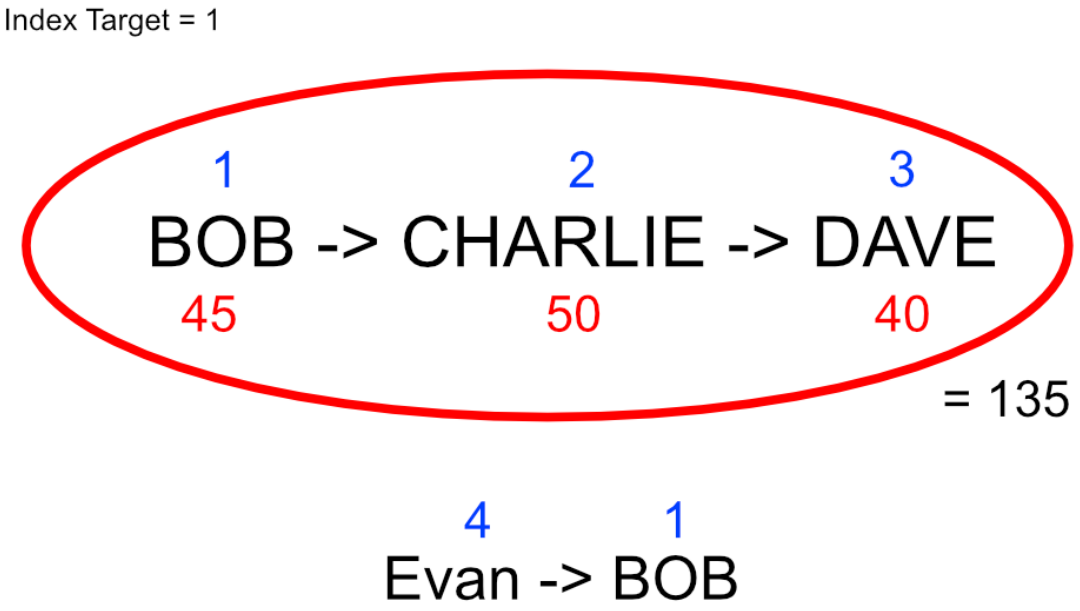
- ***STORE THE OTHER PERSON CONNECTION INSIDE EACH PERSON STRUCT (BETTER USE POINTER)***
- ***IT IS GUARANTEED THAT THERE IS NO CYCLE CONNECTION***

Examples

Input
5 Alice 60 Bob 45 Charlie 50 Dave 40 Evan 55 3 1 2 2 3

4 1 1
Output
135

Explanation



HINT

1. Add The person relation to Struct

```
struct Person {
    char name[32];
    int strength;
    struct Person *friend;
};
```

2. Set each initial friend to NULL

```
people[i].friend = NULL;
```

3. Traverse to each person

```
// TRAVERSE TO EACH PERSON CONNECTION
struct Person* temp = &people[id];
while(temp!=NULL) {
    temp = temp->friend;
}
```