

Talent Battle 100 days coding series

Sridhar was a seasoned traveler. He liked to visit new places. More than all he was a meticulous planner. This time he was planning to visit Europe. He wrote down his travel itinerary like as follows:
If he wanted to visit Madrid, Paris, Munich, Warsaw and Kiev in this order, he would write it down like as:

Madrid Paris 100

Paris Munich 200

Munich Warsaw 150

Warsaw Kiev 120

More formally, if he wanted to go from **A** to **B** directly and the price is **C** dollars, then he would write

A B C

on a card. Each move was written on a different card. Sridhar was a great planner, so he would never visit the same place twice. Just before starting his journey, the cards got shuffled. Help Sridhar figure out the actual order of the cards and the total cost of his journey.

Input Format

The first line of the input contains an integer **T**, the number of test cases. **T** test cases follow. Each case contains an integer **N**, the number of cities Sridhar is planning to visit. **N-1** lines follow. Each line is of the form

$A_i B_i C_i$

where the **i-th** line refers to the **i-th** card after getting shuffled.

Output Format

For each case the output contains **N** lines, the first **N-1** lines should contain the **N-1** cards in their proper original order, the **N-th** line should contain the total cost of the travel. See Example for detailed format.

Constraints

$$1 \leq T \leq 10$$

$$1 \leq N \leq 5000$$

$$1 \leq \text{length of } A_i \leq 50$$

$$1 \leq \text{length of } B_i \leq 50$$

$$1 \leq C_i \leq 1000$$

A_i, B_i will contain only lowercase and uppercase latin characters, no two cities will have same names. The names of cities are case-sensitive. So "warsaw" and "Warsaw" should be considered as different cities.

Talent Battle 100 days coding series

Sample Input

```
1
5
Warsaw Kiev 120
Madrid Paris 100
Munich Warsaw 150
Paris Munich 200
```

Sample Output

```
Madrid Paris 100
Paris Munich 200
Munich Warsaw 150
Warsaw Kiev 120
570
```

C++

```
#include <bits/stdc++.h>
using namespace std;

int main()
{
    int t, n;
    cin >> t;
    while (t--)
    {
        cin >> n;
        map<string, pair<string, string>> p;
        string s1, s2, s3, s4;
```

TalentBattle

Talent Battle 100 days coding series

```
set<string>s;
int x, cost = 0;
for(int i = 1; i < n; i++)
{
    cin >> s1 >> s2 >> s3;
    x = stoi(s3.substr(0, s3.length() - 1));
    p[s1] = {s2, s3};
    cost += x;
    if (s.find(s1) == s.end())
        s.insert(s1);
    else
        s.erase(s1);
    if (s.find(s2) == s.end())
        s.insert(s2);
    else
        s.erase(s2);
}
s3 = ((p.find(*s.begin()) != p.end()) ? *s.begin() : *s.rbegin());

while (p.find(s3) != p.end())
{
    cout << s3 << " " << p[s3].first << " " << p[s3].second << "\n";
    s3 = p[s3].first;
}
cout << cost << "$\n";
}
return 0;
}
```

Talent Battle 100 days coding series

Java

```
import java.io.BufferedReader;
import java.io.IOException;
import java.io.InputStreamReader;
import java.util.ArrayList;
import java.util.HashMap;
import java.util.Iterator;
```

```
public class Main {
```

```
    public static void main(String[] args) throws NumberFormatException, IOException
    {
```

```
        BufferedReader br=new BufferedReader(new InputStreamReader(System.in));
```

```
        int t=Integer.parseInt(br.readLine());
```

```
        while(t-->0)
```

```
        {
```

```
            int n=Integer.parseInt(br.readLine());
```

```
            HashMap<String,String> start=new HashMap<String,String>();
```

```
            HashMap<String, String> end=new HashMap<String, String>();
```

```
            ArrayList<Integer> cost=new ArrayList<Integer>();
```

```
            int i;
```

```
            String[] str;
```

```
            int total=0;
```

```
            for(i=0;i<n-1;i++)
```

```
            {
```

Talent Battle 100 days coding series

```
str=br.readLine().split(" ");
start.put(str[0],str[1]);
end.put(str[1],str[2]);
total=total + Integer.parseInt(str[2].replace("$", ""));
}
```

```
String costAns=total+"$";
```

```
String startCountry="";
```

```
Iterator<String>it=start.keySet().iterator();
while(it.hasNext())
{
    String c1=it.next();
    if(!end.containsKey(c1))
    {
        startCountry=c1;
        break;
    }
}
```

```
for(i=0;i<n-1;i++)
{
    System.out.println(startCountry+" "+start.get(startCountry)+"
"+end.get(start.get(startCountry)));
    startCountry=start.get(startCountry);
}
System.out.println(costAns);
```

```
}
```

```
}
```

```
}
```

Python

```
t = int(input())
while t:
    t -= 1
    n = int(input())
    forwardMap = {}
    totalCost = 0
    inData = [None]*n
    destinations = set()
    sources = set()
    for i in range(n-1):
        inData[i] = input()
        forwardMap[inData[i].split()[0]] = inData[i]
        sources.add(inData[i].split()[0])
        destinations.add(inData[i].split()[1])
        totalCost += int(inData[i].split()[2][-1])
    start = sources.difference(destinations).pop()
    for i in range(n-1):
        destination = forwardMap[start]
        print(destination)
```

Talent Battle 100 days coding series

```
start = destination.split()[1]  
print('%d$'%totalCost)
```



TalentBattle