

题目描述：

一个 XX 产品行销总公司，只有一个 boss，其有若干一级分销，一级分销又有若干二级分销，每个分销只有唯一的上级分销。

规定，每个月，下级分销需要将自己的总收入（自己的+下级上交的）每满 100 元上交 15 元给自己的上级。

现给出一组分销的关系，和每个分销的收入，请找出 boss 并计算出这个 boss 的收入。

比如：

收入 100 元，上交 15 元；

收入 199 元（99 元不够 100），上交 15 元；

收入 200 元，上交 30 元。

输入：

分销关系和收入： [[分销 id 上级分销的 id 收入], [分销 id 上级分销的 id 收入], [分销 id 上级分销的 id 收入]]

分销 ID 范围 0..65535

收入范围: 0..65535，单位元

提示：输入的数据只存在 1 个 boss，不存在环路

输出： [boss 的 ID，总收入]

输入描述：

第 1 行输入关系的总数量 N

第 2 行开始，输入关系信息，格式： 分销 ID 上级分销 ID 收入

比如：

5

1 0 100

2 0 199

3 0 200

4 0 200

5 0 200

输出描述:

输出: *boss* 的 *ID* 总收入

比如:

0 120

补充说明:

给定的输入数据都是合法的, 不存在环路, 重复的

示例 1

输入:

5

1 0 100

2 0 199

3 0 200

4 0 200

5 0 200

输出:

0 120

说明:

```
#include <bits/stdc++.h>
```

```
using namespace std;
```

```
const int N = 7e4 + 7;
```

```
int m, u, v, root, fa[N], dep[N];
```

```

bool vis[N], du[N];
long long f[N], w, va100 = 100, va15 = 15;
vector<int>son[N];
void dfs(int x){
    vis[x] = 1;
    int t = son[x].size();
    for(int i = 0;i < t; i++)
    {
        // cout<<t<<" "<<x<<" "<<son[x][i]<<endl;
        if(! vis[ son[x][i] ]) {dfs(son[x][i]);
            f[x] += f[ son[x][i] ] / va100 * va15;}
    }
}
int main() {
    cin >> m;
    for(int i = 1;i <= m; i++){
        cin >> u >> v;
        son[v].push_back(u);
        scanf("%lld", &w); f[u] = w;
        fa[u] = v; vis[u] = 1; vis[v] = 1;
        du[u] = 1;
    }
    for(int i = 0;i < N; i++)
    if(vis[i] && ! du[i]) {root = i; break;}
    memset(vis, 0, sizeof vis);
    dfs(root);
    printf("%d ", root);
    printf("%lld", f[root]);
    return 0;
}
// 64 位输出请用 printf("%lld")

```