

В рамките на тази задача ще трябва да реализирате "Студентска опашка" – структура, която максимално наподобява случващото се пред студентски стол. Когато нов студент се нарежда на опашката, вместо да застане накрая ѝ, той първо претърсва за свои "познати" в нея и ако открие такива се включва точно зад тях. Ако не намери, студентът е нямал късмет и застава накрая на опашката.

Изключването става по стандартния начин – само от началото на опашката.

**Считаме, че времето отнемащо за обслужване на един студент е 2мин, а пък нов студент се нарежда на опашката на всяка минута.** За определеност, в съвпадащите моменти първо се нарежда новият студент и чак след това първият напуска опашката. Всеки студент има име - уникален низ от главни и малки латински букви(без интервали) и уникален номер на група - цяло положително число. **Студентите с еднакъв номер на група определяме като "познати".**

### Input Format

На първия ред са зададено две числа – **N** и **M** - броят на студентите и броят на групите. Следват **N** реда с информация за всеки един от студентите. Всеки ред съдържа името и групата на текущия студент. Считаме, че те идват при опашката точно в реда, в който са въведени.

### Constraints

$$1 \leq N \leq 100000$$

$$1 \leq M \leq 100000$$

Имената на студентите са с дължина между 1 и 30 символа.

### Output Format

Програмата ви трябва да изведе на екрана точния ред, в който студентите ще излязат от опашката. За всеки студент изведете по един ред, съдържащ неговото име, времето, в което се е наредил на опашката и времето, в което е излязъл.

### Sample Input 0

```
8 666
Ivan 10
Nikolay 10
Vasil 3
Daniel 4
Yoanna 3
Maria 3
```

Pesho 666  
Gosho 10

### Sample Output 0

Ivan 0 2  
Nikolay 1 4  
Vasil 2 6  
Yoanna 4 8  
Maria 5 10  
Daniel 3 12  
Pesho 6 14  
Gosho 7 16