Quick Sort

(quick_sort.cpp/c)

Time Limit : 1 sec , Memory Limit : 131072 KB

Let's arrange a deck of cards. Your task is to sort totally n cards. A card consists of a part of a suit (S, H, C or D) and an number. Write a program which sorts such cards based on the following pseudocode:

```
Partition(A, p, r)
x = A[r]
i = p-1
for j = p to r-1
    do if A[j] <= x
       then i = i+1
           exchange A[i] and A[j]
exchange A[i+1] and A[r]
return i+1


Quicksort(A, p, r)
if p < r
   then q = Partition(A, p, r)
       run Quicksort(A, p, q-1)
       run Quicksort(A, q+1, r)
```

Here, A is an array which represents a deck of cards and comparison operations are performed based on the numbers.

Your program should also report the stability of the output for the given input (instance). Here, 'stability of the output' means that: cards with the same value appear in the output in the same order as they do in the input (instance).

Input (quick_sort.in)

The first line contains an integer *n*, the number of cards.

n cards are given in the following lines. Each card is given in a line and represented by a pair of a character and an integer separated by a single space.

Output (quick_sort.out)

In the first line, print the stability ("Stable" or "Not stable") of this output.

In the following lines, print the arranged cards in the same manner of that of the input.

Constraints

- $1 \le n \le 100,000$
- 1 ≤ the number of a card ≤ 10^9
- There are no identical card in the input

Sample Input 1

```
6
D 3
H 2
D 1
S 3
D 2
C 1
```

Sample Output 1

```
Not stable
D 1
C 1
D 2
H 2
D 3
S 3
```

Sample Input 2

```
2
S 1
H 1
```

Sample Output 2

```
Stable
S 1
H 1
```