Programming teams balanced in solidarity X71059_en

A multinational corporation participates in the CodeStrengths programming competitions with several teams of programmers every year. When organizing the teams, they always build a first team out of the best programmers from the company, next a second team out of the remaining best programmers from the company, and so on.
This year, in order to make a gesture in support of solidarity and peace amongst all the peoples of the world, the company has decided to form teams with only programmers from their staff who are from Russia and the US, so that, if possible, the teams are made out of programmers from both groups and in a balanced way.
However, the first criterion used to form a team keeps being the programming level. For example, the first team shouldn't have a programmer whose level is worse than another one who is not in the first team.
As second criterion, they will try to make each team as balanced as possible. For example, if the teams are formed by four people, then a team with two Russians and two Americans is wanted more than one with three Russians and just one American, and the latter is wanted more than a team with four Russians and no American.
As a last criterion, programmers are chosen in lexicographic order (standard comparison order between strings).

Input

The input has several cases. The first line of each case has a positive natural k, which is the number of members of each team to form. On a second line there is a positive natural n_1, which is the total number of Russian programmers. Next, n_1 lines follow, each with the name of a Russian programmer and a positive natural indicating his/her programming level. These lines are sorted from bigger to smaller programming level, and in lexicographic order for programmers with the same level. Next, there is a new line with a positive natural n_2, which is the number of American programmers. Next, n_2 lines follow, with the information about American programmers, following exactly the same format as for Russian programmers.
It is guaranteed that $n_1 + n_2$ is a multiple of k, and that there is no repetition of names. In particular, a name from one of the lists does not occur in the other list.

Output

For each case, $(n_1 + n_2)/k$ lines must be written. First line will contain the list of names of first team in lexicographic order. Second line will contain the list of names of second team in lexicographic order. And so on. After the output for each case, there is a blank line.

Sample input

```
2
3
Mikhail 7
Elena 6
Irina 3
5
Barbara 8
Jennifer 7
Karen 5
James 2
Nancy 1
4
2
Elizaveta 4
Vera 4
```

10
Betty 8
Christopher 8
Mary 6
Michael 6
John 4
Barbara 3
David 3
Patricia 2
Sarah 2
Anthony 1
5
2
Irina 5
Natalia 3
8
David 7
Barbara 6
Nancy 5
Matthew 4
Sandra 4
John 3
James 2
Richard 1
5
3
Yevgeny 7
Nikolai 5
Danyl 4
2
Barbara 1
Susan 1
3
4
Yevgeny 7
Victoria 6
Konstantin 4
Sergey 1
5
Anthony 8
Sandra 7
Susan 5
William 2
Linda 1
1
5
Sofia 5
Nikolai 4
Adelina 2
Alexander 2
Mikhail 1
8
Richard 8
Thomas 7
Patricia 5
William 5
Charles 3
Jennifer 2
Mary 1
Sarah 1
4
5
Daria 5
Xenia 5
Alexey 4
Denis 2
Maksim/Maxim 1
3
Betty 6
Lisa 6
Barbara 5
1
6
Roman 8
Anastasia 6
Irina 5
Alexey 3
Danyl 3
Natalia 3
2
Anthony 7
Sarah 7
3
3
Victoria 6
Danyl 3
Mikhail 2
6
Betty 8
Lisa 8
Patricia 7
Nancy 6
Elizabeth 3
William 3
3
2
Nadezhda 7
Sergey 5
1
William 4
5
1
Artyom 4
4
Jessica 8
Susan 6
Barbara 5
Sarah 3
1
10
Maksim/Maxim 8
Xenia 8
Elena 7
Elizaveta 7
Svetlana 7
Danyl 6
Alexander 4
Polina 4
Nikita 3
Roman 3
1
Lisa 4

```
5
5
Elena 6
Yevgeny 6
Nadezhda 3
Roman 3
Irina 2
5
Elizabeth 8
Nancy 8
Charles 6
Joseph 2
Matthew 1
3
5
Svetlana 8
Artyom 7
Sofia 6
Ivan 3
Sergey 3
4
Jessica 8
William 7
Jennifer 2
Joseph 2
2
11
Adelina 7
Elena 6
Nikolai 6
Anastasia 5
Sergey 5
Svetlana 4
Artyom 3
Mikhail 3
Yevgeny 2
Danyl 1
Ivan 1
1
John 7
5
2
Nadezhda 5
Alexander 1
3
Matthew 8
Lisa 2
Sarah 1
4
1
Arina 7
3
Lisa 2
Daniel 1
Matthew 1
4
2
Xenia 3
Dmitry 1
2
Barbara 6
John 4
1
4
Konstantin 6
Adelina 4
Alexander 3
Xenia 2
2
Charles 6
Jessica 3
3
8
Mikhail 5
Nikolai 5
Polina 5
Konstantin 4
Daria 3
Elena 3
Roman 3
Ivan 2
4
Jennifer 8
Joseph 5
Lisa 5
Sandra 2
```

Sample output

```
Barbara Mikhail
Elena Jennifer
Irina Karen
James Nancy

Betty Christopher Mary Michael
Barbara Elizaveta John Vera
Anthony David Patricia Sarah

Barbara David Irina Matthew Nancy
James John Natalia Richard Sandra

Barbara Danyl Nikolai Susan Yevgeny

Anthony Sandra Yevgeny
Konstantin Susan Victoria
Linda Sergey William

Richard
Thomas
Patricia
Sofia
William
Nikolai
Charles
Adelina
Alexander
Jennifer
Mary
Mikhail
Sarah

Betty Daria Lisa Xenia
Alexey Barbara Denis Maksim/Maxim

Roman
Anthony
Sarah
Anastasia
Irina
Alexey
Danyl
Natalia

Betty Lisa Patricia
Danyl Nancy Victoria
Elizabeth Mikhail William

Nadezhda Sergey William

Artyom Barbara Jessica Sarah Susan

Maksim/Maxim
Xenia
Elena
Elizaveta
Svetlana
Danyl
Alexander
Lisa
Polina
Nikita
Roman

Charles Elena Elizabeth Nancy Yevgeny
Irina Joseph Matthew Nadezhda Roman

Artyom Jessica Svetlana
Ivan Sofia William
Jennifer Joseph Sergey

Adelina John
Elena Nikolai
Anastasia Sergey
Artyom Svetlana
Mikhail Yevgeny
Danyl Ivan

Alexander Lisa Matthew Nadezhda Sarah

Arina Daniel Lisa Matthew

Barbara Dmitry John Xenia

Charles
Konstantin
Adelina
Alexander
Jessica
Xenia

Jennifer Joseph Mikhail
Lisa Nikolai Polina
Daria Elena Konstantin
Ivan Roman Sandra
```

Observation

Grading up to 10 points:

- Slow solution: 5 points.
- Fast solution: 10 points.

We understand as a fast solution one which is correct, with linear cost ($(n_1 + n_2)\log(k)$ is allowed in order to sort the members of each resulting team) and which passes the public

and private tests. We understand as slow solution one which is not fast, but it is correct and passes the public tests.

Problem information

Author : PRO1
Generation : 2023-12-28 23:25:48