
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 1

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

Yevgeny 6	Konstantin 6
Nadezhda 3	Adelina 4
Roman 3	Alexander 3
Irina 2	Xenia 2
5	2
Elizabeth 8	Charles 6
Nancy 8	Jessica 3
Charles 6	3
Joseph 2	8
Matthew 1	Mikhail 5
3	Nikolai 5
5	Polina 5
Svetlana 8	Konstantin 4
Artyom 7	Daria 3
Sofia 6	Elena 3
Ivan 3	Roman 3
Sergey 3	Ivan 2
4	4
Jessica 8	Jennifer 8
William 7	Joseph 5
Jennifer 2	Lisa 5
Joseph 2	Sandra 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	

Sample output 1

Barbara Mikhail	Nadezhda Sergey William
Elena Jennifer	Artyom Barbara Jessica Sarah Susan
Irina Karen	Maksim/Maxim
James Nancy	Xenia
Betty Christopher Mary Michael	Elena
Barbara Elizaveta John Vera	Elizaveta
Anthony David Patricia Sarah	Svetlana
Barbara David Irina Matthew Nancy	Danyl
James John Natalia Richard Sandra	Alexander
Barbara Danyl Nikolai Susan Yevgeny	Lisa
Anthony Sandra Yevgeny	Polina
Konstantin Susan Victoria	Nikita
Linda Sergey William	Roman
Richard	Charles Elena Elizabeth Nancy Yevgeny
Thomas	Irina Joseph Matthew Nadezhda Roman
Patricia	Artyom Jessica Svetlana
Sofia	Ivan Sofia William
William	Jennifer Joseph Sergey
Nikolai	Adelina John
Charles	Elena Nikolai
Adelina	Anastasia Sergey
Alexander	Artyom Svetlana
Jennifer	Mikhail Yevgeny
Mary	Danyl Ivan
Mikhail	Alexander Lisa Matthew Nadezhda Sarah
Sarah	Arina Daniel Lisa Matthew
Betty Daria Lisa Xenia	Barbara Dmitry John Xenia
Alexey Barbara Denis Maksim/Maxim	Charles
Roman	Konstantin
Anthony	Adelina
Sarah	Alexander
Anastasia	Jessica
Irina	Xenia
Alexey	Jennifer Joseph Mikhail
Danyl	Lisa Nikolai Polina
Natalia	Daria Elena Konstantin
Betty Lisa Patricia	Ivan Roman Sandra
Danyl Nancy Victoria	
Elizabeth Mikhail William	

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: 2026-01-25T22:08:41.794Z

© *Jutge.org*, 2006–2026.

<https://jutge.org>