

## CSCI-561 - Spring 2018 - Foundations of Artificial Intelligence

### Homework 2

**Due March 5, 2018 23:59:59**

#### Problem Description

The **2018 FIFA World Cup** will be the 21st FIFA World Cup, a quadrennial international soccer tournament contested by the men's national teams of the member associations of FIFA. It is scheduled to take place in Russia from 14 June to 15 July 2018.

The tournament organizers want to divide the qualifying national teams into a given number of groups for the 2018 FIFA World Cup. The teams within each group will all play each other, so your goal is to distribute the teams across the groups to balance capability and geography as much as possible. The qualifying teams for the FIFA World Cup come from **six continental confederations**:

1. AFC (Asia)
2. CAF (Africa)
3. CONCACAF (North and Central America)
4. CONMEBOL (South America)
5. OFC (Oceania)
6. UEFA (Europe).

To balance the groups in terms of **capability**, the teams were allocated to  **$K$  pots based on the 2017 FIFA World Ranking<sup>1</sup>**. Pot 1 contains  $N_1$  teams including Russia (which is the host for the 2018 FIFA World Cup) and the  $N_1 - 1$  highest-ranked teams in the 2017 FIFA World Ranking. Pots 2 to  $K$  contain the next  $N_i$  highest-ranked teams,  $i = 2, \dots, K$ .

Your task is to take the list of qualifying teams and assign them to groups based on which confederation and pot they are in. The draw must satisfy both of the following constraints:

- C1.** *No group can have more than one team from any pot.*
- C2.** *No group can have more than one team from any continental confederation, with the exception of UEFA, which can have up to two teams in a group.*

**There is no limit (minimum or maximum) on the number of teams in each group.**

You can use any algorithm for this homework. We have included an example of the hardest grading test case in the sample test cases. You should use the sample test

---

<sup>1</sup> The method of pot division does not matter for this HW and it has been only mentioned for the sake of completeness of problem description.

cases to optimize your program and make sure that it can handle all of them in less than **3 minutes**.

### An Illustrative Example

In order to make the problem description clearer, consider the following example with 32 national teams which are supposed to be divided into 8 groups. The classification of teams based on their continental confederations has been shown in the figure below.

<b>AFC (5)</b> <ul style="list-style-type: none"><li>•  Australia</li><li>•  Iran</li><li>•  Japan</li><li>•  Saudi Arabia</li><li>•  South Korea</li></ul>	<b>CONCACAF (3)</b> <ul style="list-style-type: none"><li>•  Costa Rica</li><li>•  Mexico</li><li>•  Panama</li></ul> <b>CONMEBOL (5)</b> <ul style="list-style-type: none"><li>•  Argentina</li><li>•  Brazil</li><li>•  Colombia</li><li>•  Peru</li><li>•  Uruguay</li></ul> <b>OFC (0)</b> <ul style="list-style-type: none"><li>• <i>None qualified</i></li></ul>	<b>UEFA (14)</b> <ul style="list-style-type: none"><li>•  Belgium</li><li>•  Croatia</li><li>•  Denmark</li><li>•  England</li><li>•  France</li><li>•  Germany</li><li>•  Iceland</li><li>•  Poland</li><li>•  Portugal</li><li>•  Russia (hosts)</li><li>•  Serbia</li><li>•  Spain</li><li>•  Sweden</li><li>•  Switzerland</li></ul>
--	--	---

Furthermore, the figure below indicates Pots 1 to 4.

Pot 1	Pot 2
Russia (65) ( <i>hosts</i> ) Germany (1) Brazil (2) Portugal (3) Argentina (4) Belgium (5) Poland (6) France (7)	Spain (8) Peru (10) Switzerland (11) England (12) Colombia (13) Mexico (16) Uruguay (17) Croatia (18)
Pot 3	Pot 4
Denmark (19) Iceland (21) Costa Rica (22) Sweden (25) Tunisia (28) Egypt (30) Senegal (32) Iran (34)	Serbia (38) Nigeria (41) Australia (43) Japan (44) Morocco (48) Panama (49) South Korea (62) Saudi Arabia (63)

The figure below shows a valid division of these teams into **8 groups**.

	1	2	3	4
Group 1	Russia	Saudi Arabia	Egypt	Uruguay
Group 2	Portugal	Spain	Morocco	Iran
Group 3	France	Australia	Peru	Denmark
Group 4	Argentina	Iceland	Croatia	Nigeria
Group 5	Brazil	Switzerland	Costa Rica	Serbia
Group 6	Germany	Mexico	Sweden	South Korea
Group 7	Belgium	Panama	Tunisia	England
Group 8	Poland	Senegal	Colombia	Japan

## File Formats

### *Input format:*

**<GROUP COUNT>**: The number of groups for the 2018 FIFA World Cup draw.

**<POT COUNT>**: The number of pots for the 2018 FIFA World Cup draw.

**<POTS DIVISION>**: It contains <POTS COUNT> lines, where the first line is a comma-separated list of the teams belonging to Pot 1, and the following lines show the teams of Pots 2 to <POTS COUNT>, respectively.

**<TEAMS CONFEDERATION>**: It contains 6 lines where each line begins with the name of one of the continental confederations (AFC, CAF, CONCACAF, CONMEBOL, OFC, or UEFA) followed by a colon ":" and then the names of the teams from this continental confederation separated by commas ",". If there is no team from a continental confederation, it is denoted by "None".

*Note: You will not be given an invalid input file. You can assume that every team that appears in the input file will be in exactly one pot and exactly one confederation. Furthermore, you will not be given any input file with <GROUP COUNT> or <POT COUNT> equal to zero.*

### *Output format:*

**<YES/NO>**: A single line containing "Yes" or "No" to indicate whether or not there is a solution for this instance of the 2018 FIFA World Cup draw. If there is a solution, output "Yes" in the first line; otherwise, output only a single line "No", with nothing else in the output file.

**<A SOLUTION>**: If there is a solution, you need to provide just one of the possible solutions. (Note that there may be more than one possible solution, but your task is to provide only one of them). In this case, after the first line (which is "Yes"), you need to output <GROUP COUNT> number of lines, where the first line indicates the names of teams for group 1 separated by commas "," and so on for groups 2 to <GROUP COUNT>. If there is no team in a specific group, you should output "None" for the line corresponding to that group.

*Note: there should not be any empty new line after the last line in the "output.txt" file generated by your code.*

## Grading

Your homework submission will be tested as follows: Your program should take no command-line arguments. It should read a text file called “input.txt” in the current directory that contains a problem definition. It should write a file “output.txt” with your solution. The formats for files “input.txt” and “output.txt” are specified below. End-of-line convention is Unix (because Vocareum is a Unix system).

During grading, 50 test cases will be used. If your homework submission passes all 50 test cases, you receive 100 points. For each test case that fails, you will lose 2 points.

Note that if your code does not compile, or somehow fails to load and parse input.txt, or writes an incorrectly formatted output.txt, or no output.txt at all, or OuTpUt.TxT, you will get zero points.

## Sample Test Cases

You are provided with 7 sample test cases and outputs. Please find them in the Homework 2 folder.

## Homework Rules

1. You must use **Python 2.7** to implement your homework assignment. You are allowed to use standard libraries only. You have to implement any other functions or methods by yourself.
2. You need to create a file named “hw2cs561s2018.py”. When you submit the homework on labs.vocareum.com, the following commands will be executed:  
`python hw2cs561s2018.py`
3. Your code must create a file named “output.txt” and print its output there. For each test case, the grading script will put an “input.txt” file in your work folder, runs your program (which reads “input.txt”), and check the “output.txt” file generated by your code. The grading script will replace the files automatically, so you do NOT need to do anything for that part.
4. Homework should be submitted through Vocareum. Homework submitted through email, in person during office hours, or through any channel other than Vocareum will not be accepted. Please only upload your code to the “/work” directory. Don’t create any subfolder or upload any other files. Please refer to <http://help.vocareum.com/article/30-getting-started-students> to get started with Vocareum.
5. Your program should handle all test cases within a reasonable time. A maximum runtime of **3 minutes** per test case will be set on Vocareum.

6. You are strongly recommended to submit at least two hours ahead of the deadline, as the submission system around the deadline might be slow and possibly cause late submission. **Late submissions will not be graded.**