Important Contest Instructions!!

Please *read the following instructions carefully. They contain important information on how to run your programs and submit your solutions to the judges. If you have any questions regarding these instructions, please ask a volunteer before the start of the competition.*

Program Input
Most programs will require input. You have two options:

1) Your program may read the input from a file. The input data will be in the local directory in the file **probXX.txt**, where 'XX' is the problem number.

2) Your program may read the input from the keyboard (standard in). You may type everything on the keyboard, or you may copy the data from **probXX.txt** into the standard in. ***Tip:*** *Type 'Ctrl-Z <return>' to signal the end of keyboard input.*

   Note: An easy way to enter keyboard data is by redirecting the contents of a file to your program. For example, if you are executing prob01, the input file **prob01.txt** can be redirected to the standard in of your program using syntax like this (examples are shown for each of the allowed languages):

   ```
   %> java prob01 < prob01.txt
   %> java –jar js.jar prob01.js < prob01.txt
   %> python prob01.py3 < prob01.txt
   %> prob01.exe < prob01.txt
   ```

   Your program will behave exactly as if you were typing the input at the keyboard.

Program Output
All programs must send their output to the screen (standard out, the default for any print statement).

Submitting your Programs

   Interpreted Programs (Java, JavaScript, Python.) Your program must be named probXX.java / probXX.js / probXX.py2 / probXX.py3, where 'XX' corresponds to the problem number. For Python, use the extension that matches the Python version you are using. Please submit only the source (.java, .js, .py2 or .py3). For java, the main class must be named probXX. Note there is no capitalization. All main and supporting classes should be in the default (or anonymous) package.

   Native Programs (C, C++, etc.) Your program should be named probXX.exe, where 'XX' corresponds to the problem number.

You are <u>strongly</u> encouraged to submit solutions for Problems #0 and #1 (see next pages) <u>prior</u> to the start of the competition to ensure that your build environment is compatible with the judges' and that you understand the Input and Output methods required.

Summary

The sole purpose of this problem is to allow each team to submit a test program to ensure the programs generated by their computer can be judged by our judging system. Your task for this program is a variation on the classic "Hello World!" program by saying hello to our two newest CodeWars sites, Barcelona and Newcastle. All you have to do is print "Welcome, Barcelona and Newcastle!" to the screen.

Output

```
Welcome, Barcelona and Newcastle!
```

> **NOTE** – this is the 2nd of two problems that can be solved and submitted before the start of the CodeWars competition. Teams are **strongly** encouraged to submit this problem **prior** to the start of the competition – hey, it's basically a free point!

Summary
You'll have no chance to win at HP CodeWars (or life) if you don't know how to do Input and Output properly. You also won't do well at CodeWars if you are rude to your judges.

Write a program to greet your esteemed judges appropriately. Read in the name of a judge and output your greeting in the appropriate format.

If you're confused at this point, go back and re-read your contest instructions.

Input
The input will be your judge's first name, a single word with no spaces:

```
Simon
```

Output
Welcome your judge with a friendly, creative greeting of some sort that includes the judge's name (does not have to match the below example):

```
Greetings, Simon the Great!  I genuflect in your general direction!
```