Sort 3 Numbers: Challenge 1

This is a **challenge** exercise. It is **not compulsory**, and may be completed **individually or with your lab partner**.

Write a program in a file called `sort3Challenge1.c` to read in 3 whole numbers with the message `Enter 3 numbers:` in the form ***`n n n`*** and print them out in ascending (non decreasing) order, one per line.

This version has a few restrictions:

- You are **not** allowed to use `if` statements.
- You are **not** allowed to use loops (e.g. `while`).
- You are **not** allowed to define functions.
- You are **not** allowed to call functions other than `printf` and `scanf` . For example, you are not permitted to use functions from the math library.
- You can use `printf` to print the value of an expression, in other words you can have an expression inside `printf` .
- You are only permitted to use parts of C covered in the week 1 and 2 lectures (except for `if` statements). For example, you are not permitted to use the ternary `?:` operator but you are allowed to use operators such as `&&` and `||` .

This is more puzzle than a programming exercise.

Try to invent your own solution - don't google or ask others how do it. Tutors will reward interesting, original or weird approaches with a Mars bar, even if they don't work.

Some Examples

```
Enter 3 numbers: 8 5 9
5
8
9
```

```
Enter 3 numbers: 0 3 5
0
3
5
```

```
Enter 3 numbers: -1 -2 -3
-3
-2
-1
```

```
Enter 3 numbers: 9 8 9
8
9
9
```

```
Enter 3 numbers: 3 2 1
1
2
3
```

To run some simple automated tests:

```
$ 1511 autotest sort3Challenge1
```

To run Styl-o-matic:

```
$ 1511 stylomatic sort3Challenge1.c
Looks good!
```

You’ll get advice if you need to make changes to your code.

Submit your work with the *give* command, like so:

```
$ give cs1511 wk03_sort3Challenge1
```

Or, if you are working from home, upload the relevant file(s) to the `wk03_sort3Challenge1` activity on Give Online.