

# Mathematics u11601 (C Programming) Michaelmas 2021

October 20, 2021

## Fourth assignment, due 12 noon, Wednesday 3/11/21

**Due after Reading Week.** Please excuse the severity of the next two paragraphs.

**Plagiarism.** This assignment is not as easy as the first three, but you have an extra week to do it. The plagiarism policy is as always: you will not copy another student's assignment. If copying is detected, all students involved will lose marks, irrespective of who copied from whom.

**Read this carefully.** You should form the habit of reading specifications carefully, and following them.

**The assignment is** to write a C program, check that it works, and submit the C program.

**Program:** To read double-precision numbers from input, using `scanf()`, to store them in an array, to store their *ranks* in another (integer) array, to use this rank information to store them in sorted order in a third array, and to print in three columns (i) the numbers in the order read, (ii) The corresponding ranks, and (iii) the same numbers in sorted order.

The arrays should be of size 1000 each: that should be plenty for this exercise.

To explain rank:

`n = 9, the number of numbers read`

number	3	1	4	1	5	9	2	6	5
rank	3	0	4	1	5	9	2	7	6

The rank of a number stored as `a[j]` is the number of indexes  $i$ ,  $0 \leq i \leq n-1$ , such that  $i \neq j$  and either  $a[i] < a[j]$  or  $(a[i] == a[j] \text{ and } i < j)$ .

Example:

```
% a.out
3 1 4 1 5 9 2 6 5
<Ctrl-D>
Input      rank      sorted
3.000000    3      1.000000
1.000000    0      1.000000
4.000000    4      2.000000
1.000000    1      3.000000
5.000000    5      4.000000
9.000000    8      5.000000
2.000000    2      5.000000
```

6.000000	7	6.000000
5.000000	6	9.000000

Edit, compile, and run your program to make sure it works. When it works correctly, submit it. Submit it using `submit-work`, which runs on `hamilton` and `synge` and probably on other maths machines.

**Points to note.**

The same remarks hold as in the previous assignments, especially that interactive input/output like

Please type in the next number:

should ***not*** occur.