

# UMB CS 310 Homework 5, Spring 2017

## Due: 12 PM (noon) on Thursday, March 30

Start with the sample C code, and implement three different methods of generating all  $N!$  permutations of  $N$  elements in Java.

Compare the performance of `permute1` and `permute3` for  $N = 8, 9, 10, \dots$ . Be very careful – permutation runtime is  $O(N!)$ . It goes through the roof before you know it.

**\*\*\* Don't run your code for more than one minute on the departmental servers. \*\*\***

You may do performance comparisons on your own machine – just say so in `ReadMe.txt`. Note that if the generated permutations are printed to the screen, runtime will be dominated by I/O. Thus to really compare performance, you should turn off the output in `printIt()` by commenting out the print statements.

Prepare a `ReadMe.txt`. Explain what you have done and special features of your code. Give some numbers for performance comparisons. Which permute method is faster? Can you explain why it is faster?

Create a folder `Homework5` under the CS 310 folder in your Unix home directory, and deposit all files in there before 12 PM (noon) on the due date. Late submission gets zero.

If you are interested in the theory behind the permute algorithms, here is a reference:  
<https://www.cs.princeton.edu/~rs/talks/perms.pdf>