Riley Payung

CDS 251

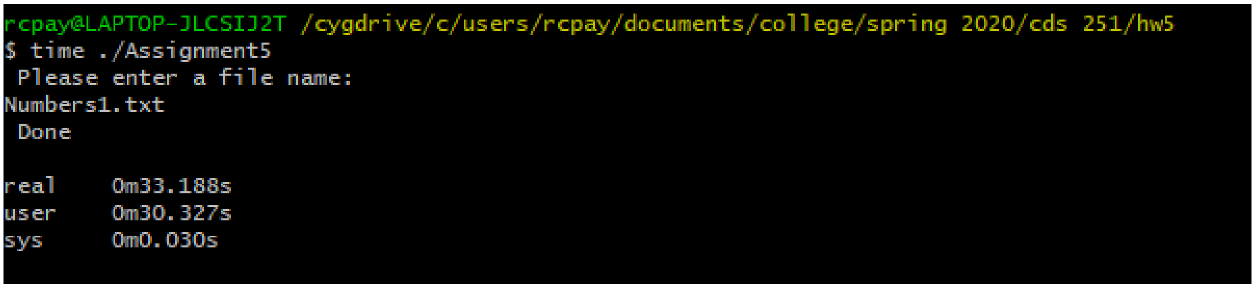
02/27/2020

Assignment 5

Implementing Bubble Sort with Indexing

I didn’t really find it hard, since I had done it before in a multitude of other programming languages. It worked on the third try, but only because I was setting j = i as the iterator in the do loop, which was dumb, since we want to go through all of the elements in the array essentially twice. A simple change of j = 1 fixed the issue.

Extra Credit

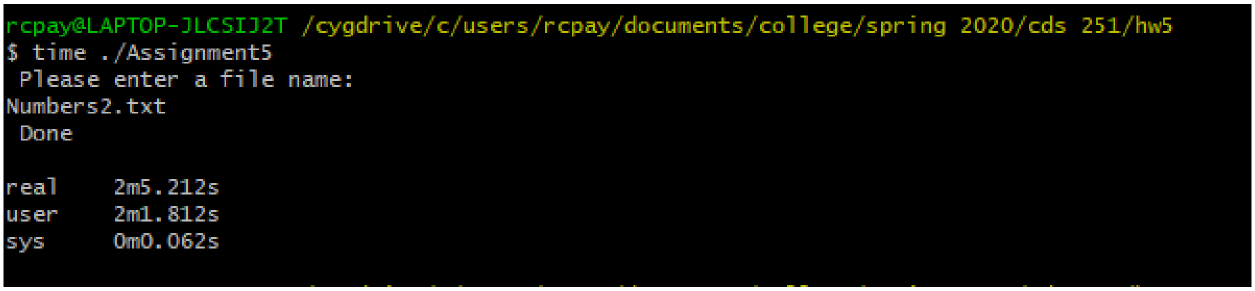


```
rcpay@LAPTOP-JLCSIJ2T /cygdrive/c/users/rcpay/documents/college/spring 2020/cds 251/hw5
$ time ./Assignment5
 Please enter a file name:
Numbers1.txt
 Done

real    0m33.188s
user    0m30.327s
sys     0m0.030s
```

The User time for the first file is 30.327 seconds.

Based on the way that the Numbers1.txt ran, the Big-O would be $O(n^2)$ and the time it **should** take to do this for 100,000 numbers should be about 4x as long, since we need to account for an extra 40,000 numbers *twice*.



```
rcpay@LAPTOP-JLCSIJ2T /cygdrive/c/users/rcpay/documents/college/spring 2020/cds 251/hw5
$ time ./Assignment5
 Please enter a file name:
Numbers2.txt
 Done

real    2m5.212s
user    2m1.812s
sys     0m0.062s
```

The user time for the second file is 2 minutes 1.812 seconds. (about 4x longer)

This shows that as the number of elements increases, the amount of time taken to sort is much much longer.