

CSCA48 Tutorial 6 - Sorting Madness

Tabeeb Yeamin, github.com/tabeebyeamin

February 27, 2020

Agenda

- Sorting Madness

Please go to:

http://www.cs.utoronto.ca/~strider/A48_programs/sorting_madness.c

And download the starter code!

- **YOUR TASK** is to update this program to work with an array of integers instead
- be careful: The random array-filling produces numbers in $[0,1]$, if you just cast that to `(int)` you'll end up with an array filled with zeros!

QSort()

- Comment out the printing statements
- Make $N=10000$, then 1000000, then 10000000, and then 100000000
- TIME how long it takes for `qsort()` to handle these inputs (in unix, this is easy, just use the `time` command:)

```
$ time ./a.out
```

Done!

```
real    0m32.278s
```

```
user    0m31.894s
```

```
sys     0m0.368s
```

(that's 100,000,000 random double numbers on Paco's 2012 laptop)

So it's really very quick!

QuickSort()

- initialize the array so it produces a *sorted* array - a common misconception is that `qsort()` is slow on an input that is already sorted. FALSE! so check it out just so we can put this false rumour to rest!

```
$ time ./a.out
```

```
Done!
```

```
real    0m7.229s
```

```
user    0m6.883s
```

```
sys     0m0.324s
```