CSC207.01 2013F, Class 37: Pause for Breath: Sorting

Overview

- Preliminaries.
  - Admin.
  - Questions on HW9.
- A few small things.
  - Formatting issues.
  - Some comments on git and github.
  - Code you should never write.
- indexOfSmallest.
- Partition for Quicksort.
- Iterative merge sort.

Preliminaries

Admin

- I returned HW8 last night. Most of today will be going over issues from that assignment, particularly looking at the two hard sorting algorithms.
- Reminder: I get so much email that sometimes I miss some, and often I file it automatically without reading it. If you need me to read your email
  - Give it a clear title that indicates that I should read it. "Homework" and "Exam" suggest to me that you are submitting work.
  - Remind me if I haven’t responded in a day or two.
  - And remind me again if I still haven’t responded.
- There are three sets of code
  - The stuff on github: [https://github.com/Grinnell-CSC207/sorting](https://github.com/Grinnell-CSC207/sorting)
  - An interactive illustration that I’ve built but didn’t document: examples/sorting-illustration
  - The things we’ll build today: examples/sorting-pause
- I will need a 207 grader for next semester. Does anyone think you can do the same level of commenting (although perhaps a bit more politely)?
- Upcoming extra credit opportunities:
  - CS Table, Today, Network Time Protocol
  - Careers in Math, Monday at noon
  - Monday, 4:15/4:30: The New CS Curriculum
  - Town Hall, Wednesday, November 13, noon or 7:30 p.m.
  - Learning from Alumni, Thursday: Atul Gupta, Trustee
  - CS Extra, Thursday: Hilary Mason ’00
  - CS Table, Next Friday, HCI
Questions on HW9

- Can we just look at powers of two and interpolate? Yes
- The RPN calculator is REQUIRED!

Some comments on git and github

- Commit regularly!
- Commit small chunks. (You can more easily undo small chunks than big ones.)
- Write decent log messages.

Formatting issues

Why might the following upset me?

```java
int indexOfSmallest(T[] vals, Comparator<T> order, int lb, int ub) {
    Integer index;
    Integer minIndex = lb;
    for ( index = lb + 1; index < ub; index++){
        if ( order.compare(vals[minIndex], vals[index]) > 0){
            minIndex = index;
        }
    }
    return minIndex;
} // indexOfSmallest(T[], Comparator<T>, int, int)
```

Code you should never write

What should happen to a 161 student who writes the following at the end of the semester?

```c
char *str = (char *) malloc(12 * sizeof(char));
str = "Hello world";
```

Important lesson: In C, when you assign to a pointer, you change what it points to.

Related lesson: In Java, when you assign to an object, you’re just assigning a pointer (although we call it a reference).

```java
String a = "Hello"; String b = "Goodbye"; String c = a; a = b;
```

What’s the relation to that and the following Java?
indexOfSmallest

What’s wrong with the following code?

```java
int indexOfSmallest(T[] vals, Comparator<T> order, int lb, int ub) {
    int i = 0;      // Fix i = lb
    while(i<vals.length)
    {
        if (order.compare(vals[i], vals[lb]) < 0) { // Fix i = ub
            lb = i;
        }
        i++;
    }
    return lb;
} // indexOfSmallest(T[], Comparator<T>, int, int)
```

What happens if we use this for selection sort?

```
5, 1, 6, 2, 8, 3
  *
1, 5, 6, 2, 8, 3
  ! * 
5, 1, 6, 2, 8, 3
  ! ! *
```

Partition for Quicksort

The code we develop can be found in examples/sorting-pause

- We’ll write invariants for partition.
- Then we’ll see an example of partition in action.
- Then we’ll write partition.

Iterative merge sort

The code we develop can be found in examples/sorting-pause

- We’ll write invariants for merge.
- Then we’ll write merge.
- Then we’ll write iterative merge sort.

Copyright (c) 2013 Samuel A. Rebelsky.