CSE 12 – Basic Data Structures and Object-Oriented Design
Lecture 12

Greg Miranda & Paul Cao, Winter 2021

This lecture is being recorded

Announcements

- Quiz 12 due Friday @ 8am
- Survey 5 due Friday @ 11:59pm
- PA4 due tonight @ 11:59pm
- PA5 released tomorrow
  - Closed for collaboration

Topics

- Questions on Lecture 12?
- Combine/Sort

Questions on Lecture 12?

SortFast – Draw the picture of sortC()

```
public class SortFast {

 public static String s(int[] arr) {return Arrays.toString(arr); }

 public static int[] combine(int[] part1, int[] part2) {
  int index1 = 0, index2 = 0;
  int[] combined = new int[part1.length + part2.length];
  while(index1 < part1.length && index2 < part2.length) {
   if(part1[index1] < part2[index2]) {
    combined[index1 + index2] = part1[index1];
    index1 += 1;
   }
   else {
    combined[index1 + index2] = part2[index2];
    index2 += 1;
   }
  }
  while(index1 < part1.length) {
   combined[index1 + index2] = part1[index1]; index1 += 1;
  }
  while(index2 < part2.length) {
   combined[index1 + index2] = part2[index2]; index2 += 1;
  }
  System.out.println(s(part1) + " + " + s(part2) + " -> " + s(combined));
  return combined;
 }
```

```
 public static int[] sortC(int[] arr) {
  if(arr.length <= 1) {return arr; }
  else {
   int[] part1 = Arrays.copyOfRange(arr, 0, arr.length / 2);
   int[] part2 = Arrays.copyOfRange(arr, arr.length / 2, arr.length);
   System.out.println(s(arr) + " -> " + s(part1) + " + " + s(part2));
   int[] sortedPart1 = sortC(part1);
   int[] sortedPart2 = sortC(part2);
   int[] sorted = combine(sortedPart1, sortedPart2);
   return sorted;
  }
 }
}


main() {
 int[] result = SortFast.sortC(new int[]{34, 93, 12, 49, 69, 25, 39 });
 System.out.println(SortFast.s(result));
}
```