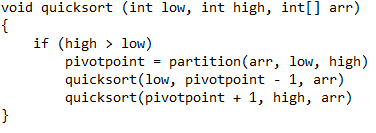
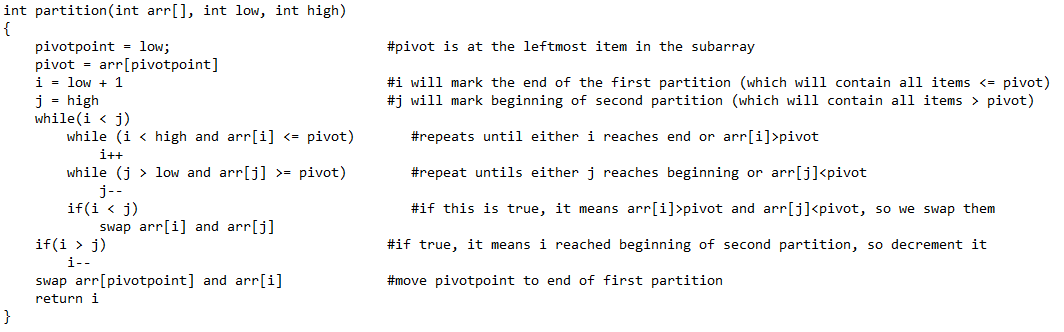
**CIS 360 Lab #8 Divide and Conquer: Quick Sort**

**Name \_\_\_Tanmay Sonawane\_\_\_\_\_ ID \_\_\_01996456\_\_\_\_\_\_\_\_\_\_\_**

1. Implement the *QuickSort* algorithm (pseudocode is below), as well as the *Partition* algorithm that it utilizes to partition the array.





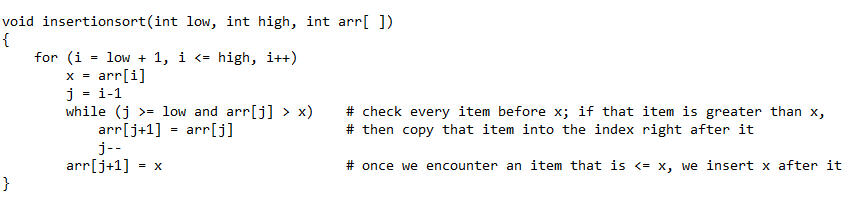
1. In the above algorithm, the first item in the list is always selected as the pivot item. Now, implement a new version of *Partition*, entitled *Partition\_With\_Median*, that uses “the median of the three”: this should assign the pivot to be the median of the first, the middle and the last item in the list.

For example, given the following list: {56, 34, 189, 240, 150, 12, 9, 123}

The first item is 56, the last is 123, and the middle item (*calculated as low+high/n*) is 240. The median of these three numbers is 123, therefore the pivot is 123.

To test this program, you can use the original *QuickSort* algorithm, except you must change it to call *Partition\_With\_Median* rather than the original *Partition* method.

1. Create a new version of *QuickSort*, entitled *QuickSort\_With\_Cutoff*. This should be structurally identical to the original except that, after calling *Partition*, it should check if either of the partitions have a length <= 10; if so, it should call *InsertionSort* (see pseudocode below) for that partition, but if not, it should recursively call itself for that partition (like in the original *QuickSort*).



1. Run experiments for each of the four different techniques, using randomly-generated arrays of various sizes (see table below). Use a **counter variable** to count the number of times that *QuickSort* recursively calls itself and/or *InsertionSort*, and record these runtimes in the table below.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **n=100** | **n=5000** | **n=50,000** | **n=100,000** |
| **QuickSort( ) and Partition( )** | **133,66** | **6948,3438** | **74663,** **35510** | **210280,** **84521** |
| **QuickSort( ) and Partition\_with\_median( )** | **131,65** | **6746,3308** | **73035, 33145** | **206140,** **66553** |
| **QuickSort\_with\_cutoff( ) and Partition( )** | **16,17** | **865,852** | **10092,** **9230** | **35085,** **24996** |
| **QuickSort\_with\_cutoff( ) and Partition\_with\_median( )** | **14,17** | **822,821** | **8908,8307** | **25140,16689** | |