

**COE 205, Term 092**  
**Computer Organization & Assembly Programming**

**Programming Assignment# 2**  
**Due date: Monday, May 3, 2010**

**Q.1.** The quick sort algorithm sorts by employing a [divide and conquer](#) strategy to divide a [list](#) into two sub-lists.

The steps are:

1. Pick an element, called a *pivot*, from the list.
2. Reorder the list so that all elements with values less than the pivot come before the pivot, while all elements with values greater than the pivot come after it (equal values can go either way). After this partitioning, the pivot is in its final position. This is called the **partition** operation.
3. Recursively sort the sub-list of lesser elements and the sub-list of greater elements.

The base case of the recursion are lists of size zero or one, which are always sorted.

In simple pseudocode, the algorithm might be expressed as this:

```
procedure quicksort(array, left, right)
  if right > left
    select a pivot index //(e.g. pivotIndex := left+(right-left)/2)
    pivotNewIndex := partition(array, left, right, pivotIndex)
    quicksort(array, left, pivotNewIndex - 1)
    quicksort(array, pivotNewIndex + 1, right)

function partition(array, left, right, pivotIndex)
  pivotValue := array[pivotIndex]
  swap array[pivotIndex] and array[right] // Move pivot to end
  storeIndex := left
  for i from left to right - 1 // left ≤ i < right
    if array[i] ≤ pivotValue
      swap array[i] and array[storeIndex]
      storeIndex := storeIndex + 1
  swap array[storeIndex] and array[right] // Move pivot to its final place
  return storeIndex
```

- (i) Write a procedure, **partition**, to implement the partition function.
- (ii) Write a procedure, **quicksort**, to implement the quicksort procedure.
- (iii) Ask the user to enter the number of integers to be sorted, n.
- (iv) Ask the user to enter an array of n integers and read it.
- (v) Use the **quicksort** procedure you implemented to sort the array, IntArray.
- (vi) Display the array, IntArray, after sorting.

A sample execution of the program is shown below:

*Enter the number of integers to be sorted: 5*

*Enter an array of 5 integers:*

*2 1 3 5 6*

*Array after sorting is:*

*1 2 3 5 6*

*This assignment is to be done in groups of two. The solution should be well organized and your program should be well documented. Submit a soft copy of your solution in a zip file. Your solution should be submitted in a word file that contains the following items:*

- i) Your names and IDs*
- ii) Assignment number*
- iii) Problem statement*
- iv) Your solution along with the code*
- v) Discussion of what worked and what did not work in your program. Include snapshots that demonstrate the working parts of your program. If things did not work and you attempted to solve them, mention that and write about the difficulty that you have faced.*

*The soft copy should also contain both source code file (i.e. .asm) and the executable file (i.e. .exe).*