COE 301 / ICS 233 – Computer Organization

MIPS Programming Assignment 2, Term 162

Due date: Thursday 20/04/2017

Quick Sort Recursive Function

The function **quick_sort** sorts an array of doubles recursively. Translate this function into MIPS code. Write a main function to allocate dynamically an array of **n** doubles (**n** is a user input that must be greater than 1), read an array of **n** doubles, print the array of **n** doubles before sorting, sort the array, and then print the array after sorting. You need to write functions for reading and printing the array, and pass parameters properly according to the MIPS convention.

```
void quick_sort(double array[], int n) {
  int i = 0;                          // i = low index
  int j = n-1;                        // j = high index
  double pivot = array[(i+j)/2];      // pivot = middle value
  while (i < j) {
    while (array[i] < pivot) i++;
    while (array[j] > pivot) j--;
    if (i < j) {
      double temp = array[i];
      array[i] = array[j];            // swap array[i]
      array[j] = temp;                // with array[j]
      i++;
      j--;
    }
  }
  if (j > 1)   quick_sort(&array[0], j);         // Recursive call 1
  if (i < n-2) quick_sort(&array[i+1], n-i-1);   // Recursive call 2
}
```

```
void read_array (int n, double array[]) {
  // Ask the user to input n elements of type double
  // The user input should be stored in array[]
}
void print_array (int n, double array[]) {
  // Display the n elements of array[]
}
```

Submission Guidelines:

All submissions should be done through Blackboard. Submit the source code of the program. Make sure that your program is well written and documented. The program will be graded according to its correctness and documentation. It is your responsibility to make sure that the program works. A program that does not assemble or run will receive zero on correctness. Copying programming assignment is not allowed. This is individual work. Detected copies will get zero grades. This includes the one who wrote the program and the one who copied it.

Grading Scheme:

Dividing the program into procedures and passing parameters properly	**[3 points]**
Allocating an array of *n* doubles dynamically	**[1 point]**
Reading the array	**[2 points]**
Sorting the array properly using recursive quicksort	**[10 points]**
Printing the array before and after sorting	**[3 points]**
Program readability and comments	**[1 point]**