CS110: Computer Programming Lab
Department of CSE
IIT, Guwahati

Module 04 Stage 01 Exercise 05

Exercise

A popular sorting algorithm based on recursive calls is called quicksort.

The students will be required to implement the algorithm as advised using only pointers to access the elements of the array a defined in function `main()`. The array should not have any copy in the program. The program should not use any indexed entry such as `data[i]`. The access should be only through the pointer variables.

Two core functions in the implementation are `split()` and `sort()`. Function `sort()` is very simple recursive function. It uses `split()` to organise a pointer into array `data` called `mid`. The pointer `mid` is so set that all entries from pointer `begin` to `mid` in array `data` are less than or equal to `*mid`. Entries in array `data` after that up to pointer `end` are greater than `*mid`.

As two parts are disjoint and all values in the second part are larger than the values in the first part, two parts can be sorted recursively to have the full list sorted.

Function `split()` may be built as follows. First set aside value at the planned value for location `mid` to be returned as return value. Typically, `*begin` serves this purpose.

All values in the segment begin+1 to last can be systematically scanned and swapped so that all values less than or equal to `*begin` are located on the left (in location `begin + 1` onwards) and all lager values are moved to the right (`last` is the last location). Pointer mid is the transition location at the junction of the two halves with `*mid` being less than or equal to `*begin`. Now, to be certain `*begin` is swapped with `*mid` before returning pointer `mid` as return value for function `split()` value.

Write codes for functions `split()` and `sort()` to complete your sorting function.

Appendix

```
#include <stdio.h>

int *split(int *begin, int *end)
{
```

```
     /* Code removed */
}

void sort (int *begin, int *end)
{    /* All other lines removed */
    int *mid;
    mid = split(begin, end);
}

int main()
{
    int data[20] = {1, 3, 5, 7, 8, 0, 9,  6,
          4, 2, 6, 0, 2, 9, 3, 2, 8, 8, 7, 7};
    int *begin, *end;

    begin = data;
    end = begin+19;

    while (begin <= end)
    {
        printf("%d ", *begin);
        begin++;
    }
    printf("\n");
    begin = data;

    sort(begin, end);

    while (begin <= end)
    {
        printf("%d ", *begin);
        begin++;
    }
    printf("\n");
}
```

Error Reporting and Suggestions for Improvements

My sincere apologies if the document has errors or mistakes. Please report errors in this document to vmm@iitg.ernet.in. Also, I welcome suggestions and advice to improve the quality of the document for the students of CS110.