ESO207: Programming Assignment 1

Due on 6 Sept, 2015. To be submitted online.

Problem In this assignment you are required to implement k-way Merge Sort algorithm.

In this version partition the input sequence of integers into k almost equal (may differ by at most 1) subsequences, recursively sort, and then merge the k sequences.

Input: A positive integer n, a sequence of integers $(a_1, a_2, \ldots, a_n)$, and a positive integer $k \geq 2$.

Goal: Design a program $KWMS(A, i, j, k)$ which sorts in decreasing order the integers contained in an array A in the index range $i : j$ (including i and j) using k-way merge. After the completion of sorting the program should print $A[i : j]$ starting from the new line: *The sorted list in the range i:j is ....*

Details: Please implement the program by strictly following these step.

1. Use three global arrays A, B, C. A contains the input sequence. B is used to form a MaxHeap, and C is used for temporary storage.

2. Let $a = \lceil (j-i+1)/k \rceil, b = \lfloor (j-i+1)/k \rfloor, r = (j-i+1)\%k$ (remainder of $(j-i+1) \div k$). Partition the array $A[i : j]$ into $A[i : i+a-1], A[i+a : i+2a-1], \ldots, A[i+(r-1)a : i+ra-1], A[i+ra : i+ra+b], A[i+ra+b : i+ra+2b], \ldots$.

3. In order to perform k-way merge implement a MaxHeap on another array B. Let B be a 2D array with the range $[0 : 1][1 : k]$. To store integer x of subarray j by setting $B[0][\alpha] = x$ and $B[1][\alpha] = j$.

4. To perform merge operation, first enter the greatest element of each non-empty sub-array into the heap, starting from the leftmost subarray (lower indices to the higher indices). Then each time the greatest element is extracted from the Heap, identify its subarray from $B[][1]$ and insert the next element from that subarray into the heap. If that sub-array becomes empty, then no insertion will occur. **MaxHeap**

must be implemented exactly the way we discussed in the class. Use $HeapSize$ to keep track of the number of elements currently in the heap.

5. The merge must be done into array C and then its content must be transferred back to A.

6. Take the array A and C lengths to be 1000 each.

7. To help us evaluate the correctness of the program, please print from a fresh line *Content of the heap is* $B[0][1], B[0][2], \ldots, B[0][k]$ after each extraction+insertion (or after extraction, if no insertion happens) in the heap.

At the end of the routine $KWMS(A, i, j, k)$ put a print statement which prints from a fresh line *The sorted list in the range* $i : j$ *is* $A[i], A[i+1], \ldots, A[j]$. Note that this being a recursive program this statement will get printed after each recursive call.