ECE-250 – Algorithms and Data Structures (Winter 2012)

Tutorial 2 (2012-01-26)

1 - Assuming well-behaved positive functions such as those that we encounter when analyzing algorithms, show that:

(a) Little-oh defines a partial ordering.
(b) Big-Oh defines a weak ordering.
(c) Big-Theta is an equivalence relation (one of the three conditions was already done during last week's tutorial session).

2 - Determine the run time (worst-case only) of the following operations:

(a) Find the first negative value in an array of:
(i) Random values
(ii) Ordered values in ascending sequence
(iii) Ordered values in descending sequence

(b) Same as part (a), but for a linked list

(c) Insert an element into a singly-linked list with random values:
(i) After a given element (i.e., you're given a pointer to a node)
(ii) Before a given element
(iii) Exactly at the middle (plus/minus rounding if the list contains an odd number of elements)

3 - Compare (a discussion suffices; no need to formally determine the actual figures) the run times (if applicable, discuss the "actual time" performance) of operations on a queue (enqueue and dequeue) when we implement it in terms of:

(a) An array (linear array)
(b) A circular array or circular buffer
(c) A linked list

4 - (If enough time) Determine the run time of the following function, operating on a polygon (a sequence of points) represented as an array of point objects (we have some class Point to represent the points/vertices of the polygon):

The function needs to check a condition for every pair of points; the condition is symmetric (i.e., if the condition is met for points p_i,p_j, then it is also met for the pair p_j,p_i), and it is checked in constant time. It is also known that the condition is never met between a point and the five points that follow.

Determine the runtime of a function meeting the above requirements.