

Assignment 5.

Post date: Nov. 4, 2014.

Due date: Nov. 10, 2014 (Monday) by noon. Hand in printed paper at assignment box at 4th floor of MC across from the tutorial centre.

Marking TA: Chen Fei Du

Mandatory requirement for assignment being accepted:

The answers need to be typed up with a computer and printed on letter size paper. Only minor corrections can be made with handwriting. The first page of the submission must be a cover page that contains your name, student id, and course number (CS466 or CS666) and assignment number.

Question 1. (10 marks)

This question is from Textbook [CLRS] 2nd edition, Exercise 17.2-3.

Suppose we wish not only to increment a counter but also to reset it to zero (i.e., make all bits in it 0). Show how to implement a counter as an array bits so that any sequence of n INCREMENT and RESET operations takes time $O(n)$ on an initially zero counter. (Hint: Keep a pointer to the high-order 1.) Write the pseudo code and proof.

Question 2. (10 marks)

In class we introduced a way to use a dynamic linked list to maintain a set of n objects that may be queried with different frequencies. Let $cost_{MTF}$ be the total cost (in terms of number of comparisons) used for a sequence of queries in a dynamic linked list maintained with the MTF (Move To Front) method. Let S_{opt} be the cost of the same sequence of queries on an optimal static linked list. If the MTF list initially contains all the objects in arbitrary order, we proved in class that $cost_{MTF} \leq 2 \cdot S_{opt} + \frac{n(n-1)}{2}$.

Now consider a variation, where the MTF list is initially empty. When querying one of the n objects, if it is in the MTF list, then the cost is calculated as usual and MTF is used to maintain the list. If the object is not in the list, you simply insert it to the front of the MTF list. However, the cost for such a query is the length of the list (because you can only realize the fact after scanning the whole list).

Under this new model, modify the proof in the class to show that $cost_{MTF} \leq 2 \cdot S_{opt}$ for a sequence of queries. Write the complete proof. Note that you are required to use the potential analysis.