

CS3230: Assignment for Week 7

Due: Sunday, 20th Mar 2022, 11:59 pm SGT.

Please upload PDFs containing your solutions (hand-written & scanned, or typed) by 20th Mar, 11:59 pm to **Assignments/Assignment7/Submissions**. Name the file **Assignment7_SID.pdf**, where SID should be replaced by your student ID.

You may discuss the problems with your classmates or read material online, but you should write up your solutions on your own. Please note the names of your collaborators or online sources in your submission; failure to do so would be considered plagiarism.

Note: For all questions in this assignment, you should justify your answer.

1. (1 point) Suppose we perform a sequence of n operations. The i -th operation costs i if i is a power of 2, and costs 1 otherwise. Determine the (asymptotic) amortized cost per operation. Answer using the Θ -notation.
2. (7 points, graded for correctness) Consider a variant of the binary increment problem where the cost of flipping the i -th bit is 2^i (the rightmost bit is considered as the 0-th bit). Determine the (asymptotic) amortized cost per operation when performing n increment operations starting from 0. Answer using the Θ -notation.
3. (1 point) In the (original) binary increment problem, suppose that we wish not only to increment a counter but also to *reset* the counter to zero (i.e., set all bits in it to 0). Considering the time to examine or modify a bit as unit time, show how to implement a counter as an array of bits so that any sequence of n INCREMENT/RESET operations starting from 0 takes time $O(n)$.

(Hint: Keep a pointer to the highest-order 1-bit, so that you do not have to spend time scanning the whole array to search for this bit.)