

Checkpoint 3 Revision

Jacob Purcell, Texas A&M, Student

I. 2

My first attempt assumed a datastructure in which the computer iterated through the list in the background in the definition of `operator[]` (some accessing operator), while the tail for a linked list has the pointer to the end stored.

Since `[]` is constant access, both would have a time complexity of $O(1)$, since the pointer to the tail will retrieve the location with $O(1)$.

II. 5

Same as part 2, I assumed a datastructure in which `[]` was an $O(N)$ operation.

Considering that `[]` is constant access, keeping the same logic for the linked list, both would have a time complexity of $O(1)$.

III. 7

Initially I assumed a datastructure in which the operator `[]` was $O(N)$.

Since `[]` is constant time, and one does not have the pointer to `k` stored, one would need to iterate through $O(N)$ a linked list to get to `k` while `[]` will return `k` with $O(1)$.

IV. 8

This one I guessed because I didn't know what the "best" search invented so far was.

After research, I found that searching a hash table has a time complexity $O(1)$.