$n$ = current size of Vector

- Simple Vector

$(C_1 + \sum_{0}^{n-1} C_2 + C_3)$ · number of insertions

$\Rightarrow O(n)$ · number of insertions

- Efficient Simple Vector

if the array is full

$(C_1 + \sum_{0}^{n-1} (C_2 + C_3)$ · number of insertions

$\Rightarrow O(n)$ · number of insertions

if the array is not full

$O(4)$ · number of insertions

- Simple Vector with Linked List

$C_1 + \sum_{0}^{n-1} C_2 + C_3$

$\Rightarrow O(n)$ · number of insertions

* I will note that $C_2$ is much smaller in Simple Vector with Linked List than in the other two Simple vectors, meaning there will be significantly fewer operations performed compared to the other two

Scanned by TapScanner