

week5 HW 戴偉璿

第一題

假設一開始的陣列都只有一格

a.

時間複雜度的部份，我們先假設 $n=2^a+b; a, b \in \mathbb{N}; b < 2^{a+1}$ ，很顯然所需的複製次數為 $\sum_{k=0}^a 2^k = 2^{a+1} - 1$ ，插入元素的複雜度為 b ，其時間複雜度為 $O(2^{a+1} + b) = O(2n) = O(n)$

空間複雜度的部份，最終所需的空間為 $O(2^{a+1}) = O(2n) = O(n)$

b.

在插入新元素時，會先新增一個比原大小多一的陣列，並且將舊有的所有資料複製過去，由於每次都只會開「剛剛好」，因此空間複雜度為 $O(n)$

但是每次都會將舊有的所有資料複製過去，在插入第二個元素時有一個元素需要複製，拆入第二個元素時需要複製兩個元素...插入第 n 個元素時需要複製 $n-1$ 個元素。所需時間複雜度為每次操作時所需複製元素數目的總和， $\sum_{i=1}^{n-1} i = \frac{(n-1)n}{2}$ ， $\text{big-O} = O(n^2)$

第二題

事先聲明： $\frac{n}{k}$ 除以 k 之後無條件捨去取至整數（跟 C++ 一樣）

a.

預處理： $O(NK)$

總詢問： $O(Q \times \frac{N}{K})$

b.

根據 2.(a) 的答案，預處理複雜度為 $O(NK)$ ，總詢問複雜度為 $O(Q \times \frac{N}{K})$ 。又 $O(N) = O(Q)$

因此 $NK = Q \times \frac{N}{K} \Rightarrow K = \sqrt{N}$

c.