Lab 04 Exercise 2
Group 7

**1. Explain the difference between an array size and capacity**

Array size is the number of indices/slots that actually contain data while capacity is just the total number of indices/slots (the length of the array).

2. What happens when an array needs to grow beyond its current capacity? Explain and produce a diagram showing the memory layout before and after expansion.
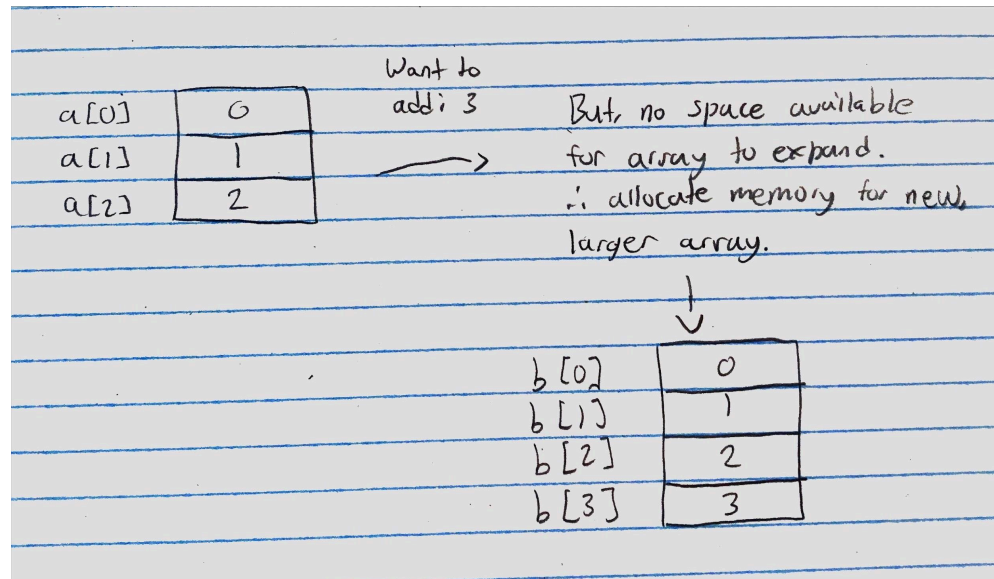
1. **First consider the case where there is space in memory after the end of the array.**

   If there is space in memory after the end of the array, we can just expand the array using that space.
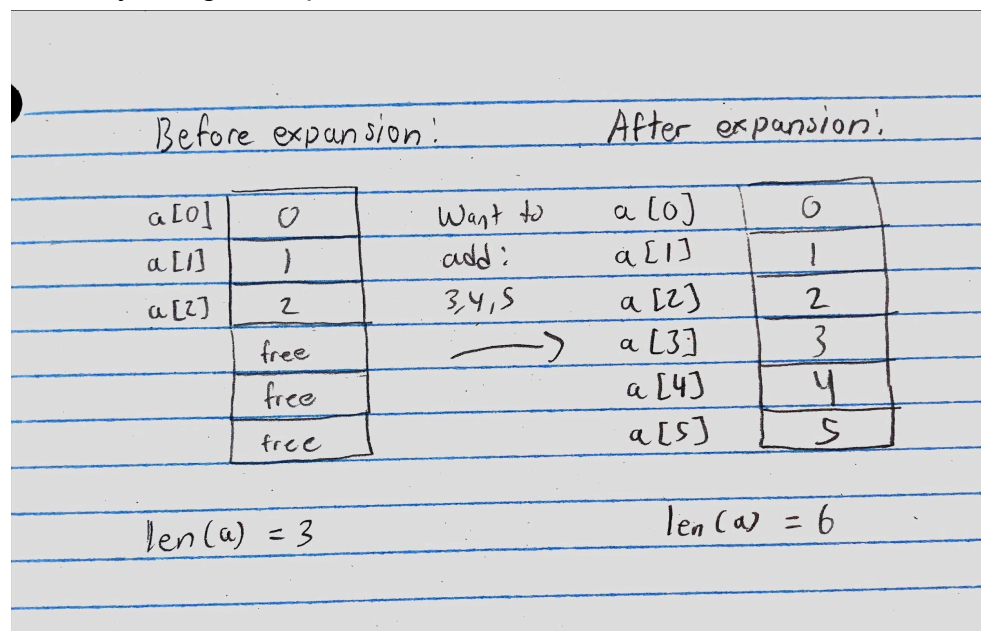
2. **Then, consider the case where the memory after the end of the array is occupied by another variable. What happens in this case?**

   If the memory after the array is occupied by another variable, we can no longer just expand the array. Instead, we need to allocate a larger block of memory for a new array, and all elements from the original array must be copied to our new array (Python's garbage collection will deal with memory allocation/deallocation).

Want to
add 3

a[0]   0
a[1]   1
a[2]   2

→

But, no space available
for array to expand.
∴ allocate memory for new,
larger array.

↓

b[0]   0
b[1]   1
b[2]   2
b[3]   3

3. Discuss one or more techniques real-world array implementations use to amortize the cost of array expansion.

One real-world technique array implementations use is to over-allocate memory when creating the array initially (like in Python). This allows future insertions into array elements that exceed the original array's capacity without having to constantly copy over arrays and reallocate memory which uses up a lot of space.