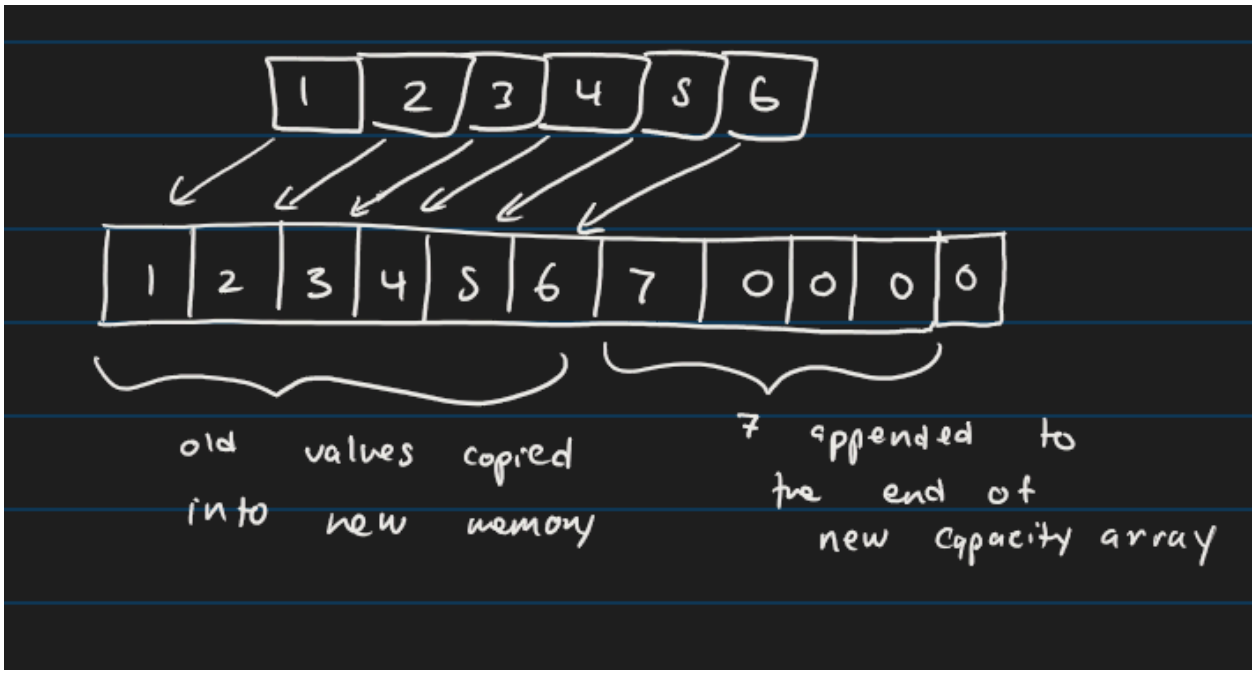
2.1

The difference between array size and capacity is that the size of the array refers to the number of elements that the array currently holds. Capacity of the array refers to the number of elements that the array can hold, an array's capacity is the allocated amount of memory for the array.

2.2

When an array needs to grow beyond its current capacity, it needs to be resized, because more memory needs to be allocated to fit the new elements added to the array.

1. In the case where there's space in memory after the array, then the array extends its capacity into the reserved space of the underlying array.
2. In the case where the memory at the end of the array is occupied by another variable, the array is resized, it allocates a new, larger chunk of memory. This chunk is larger than usually what is necessary to allow more room for any other new elements. After the larger memory is allocated, the content of the old array is copied into the newly allocated memory, with the new element at the end. The old memory is deallocated to free up the space. The array now points to the new memory chunk.



2.3

One technique that real-world array implementations use to amortize the cost of array expansion is to allocate extra capacity initially, because anticipating the needed capacity will minimize the number of allocations, which would improve the performance.