

1. Whereas a stack is LIFO, thus things are only removed/added from one end, queues are FIFO, and things are added/removed from both ends thus requiring circular array implementation.
2. The algorithm would double the size of the array and leave all elements in indexes prior to the back index alone while moving all the elements after (i.e., the elements including and after the front index) to the back of the array. This would leave space between the front and back indexes and allow the back to continue growing.
3. The amortized running time would be  $\Theta(1)$  because the total cost is always a constant.