**Queue - Implementation**

[Report Issue](https://github.com/LeetCode-Feedback/LeetCode-Feedback/issues)

To implement a queue, we may use a dynamic array and an index pointing to the head of the queue.

As mentioned, a queue should support two operations: enqueue and dequeue. Enqueue appends a new element to the queue while dequeue removes the first element. So we need an index to indicate the starting point.

Here is an implementation for your reference:

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer program

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

*Drawback*

The implementation above is straightforward but is inefficient in some cases. With the movement of the start pointer, more and more space is wasted. And it will be unacceptable when we only have a space limitation.

A blue arrow pointing to a rectangular object

AI-generated content may be incorrect.

Let's consider a situation when we are only able to allocate an array whose maximum length is 5. Our solution works well when we have only added less than 5 elements. For example, if we only called the enqueue function four times and we want to enqueue an element 10, we will succeed.

And it is reasonable that we can not accept more enqueue request because the queue is full now. But what if we dequeue an element?

A blue arrow pointing to a rectangular object

AI-generated content may be incorrect.

Actually, we should be able to accept one more element in this case.