Chapter 1

Queues

1.1 Introduction

Queues are one of the simplest data structures that one will learn about. Most likely you have heard the word **queue** in the context of a waiting line or such. The idea of a **queue** in Computer Science is really no different.

1.2 Description

Queues work in a FIFO (first-in first-out) manner. This simply means that the first thing into the **queue** will be the first thing out, or if extended, items that enter the **queue** earlier exit the **queue** earlier. To build on the waiting line example, someone who lines up earlier than you will reach the end of the **queue** and exit earlier than you.

1.3 Implementation

To support this behaviour, **queues** implement the following methods:[1]

- **enqueue**
  Enqueue is used to add elements to the back of the **queue**

- **dequeue**
  Dequeue is used to remove elements from the front of the **queue**

1.4 Examples

1. Enqueueing
   Starting with the following **queue** (arrows point towards the front), **enqueue** 3.

$$8 \rightarrow 9 \rightarrow 0$$

$$Ans: 3 \rightarrow 8 \rightarrow 9 \rightarrow 0$$

2. Dequeueing
   What do you get when you **dequeue** from the previous solution?

$$Ans: 0$$

[1] Depending on your programming language, the method names may not be accurate. Nonetheless, there should be methods that provide identical functionality