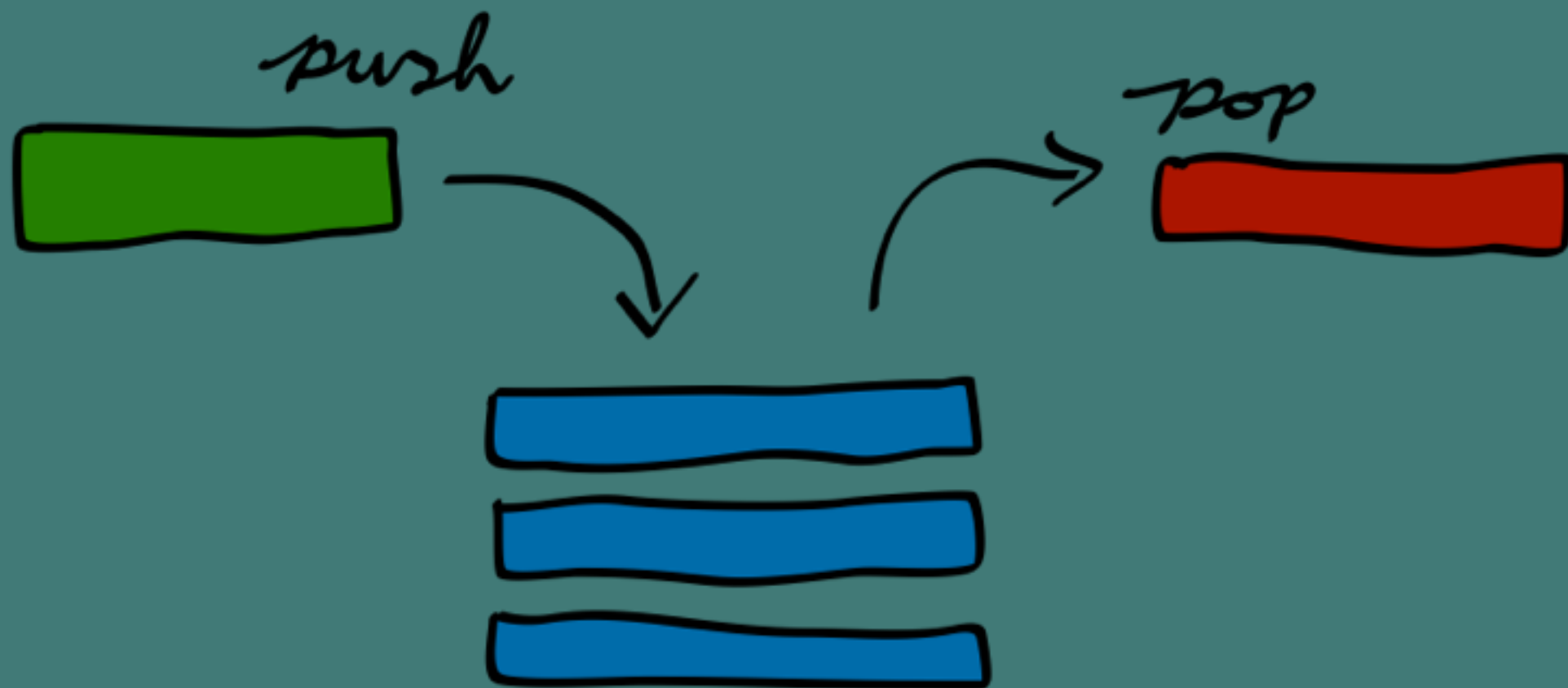# STACKS & QUEUES

WHAT IS A STACK?
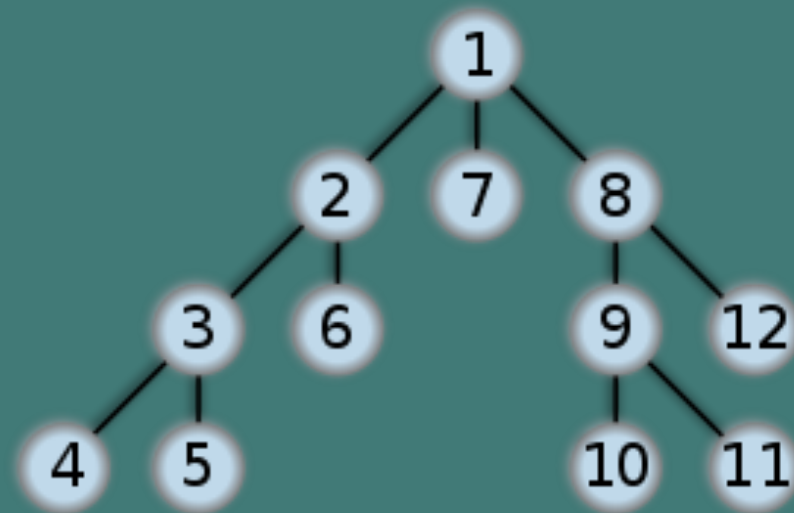
WHAT IS A STACK?

- In computer science, a stack or LIFO (last in, first out) is an abstract data type that serves as a collection of elements, with two principal operations: push, which adds an element to the collection, and pop, which removes the last element that was added.
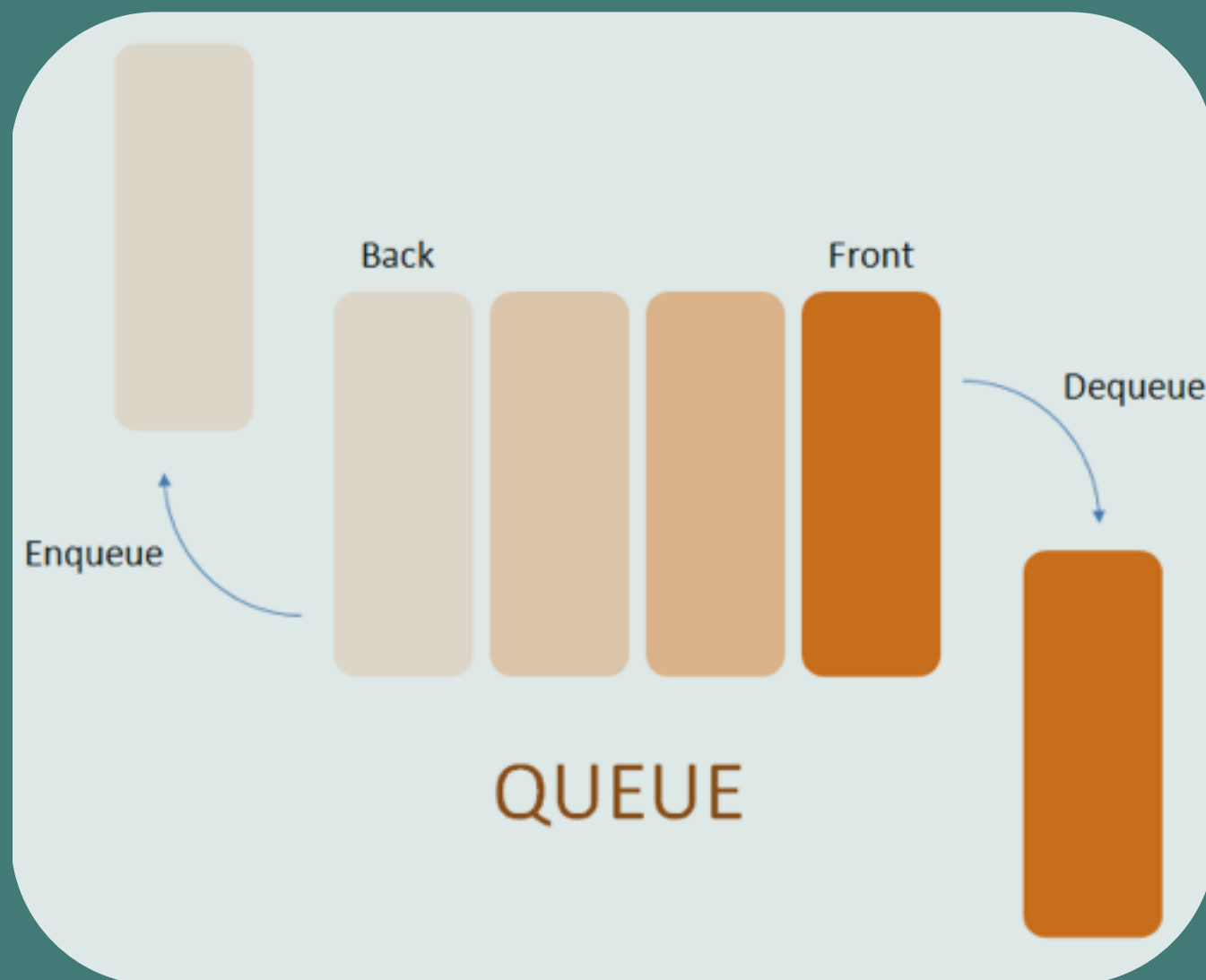
WHAT ARE STACKS USED FOR?

- Reversing strings

- Undo changes (e.g. a text editor)

- Backtracking

- Parsers

- Depth-First Search

HOW TO IMPLEMENT A STACK IN RUBY?

```ruby
class Stack
  def initialize
    @list = Array.new
  end

  def push(item)
    @list.push(item)
  end

  def pop
    @list.pop
  end

  def peek
    @list.last
  end
end
```
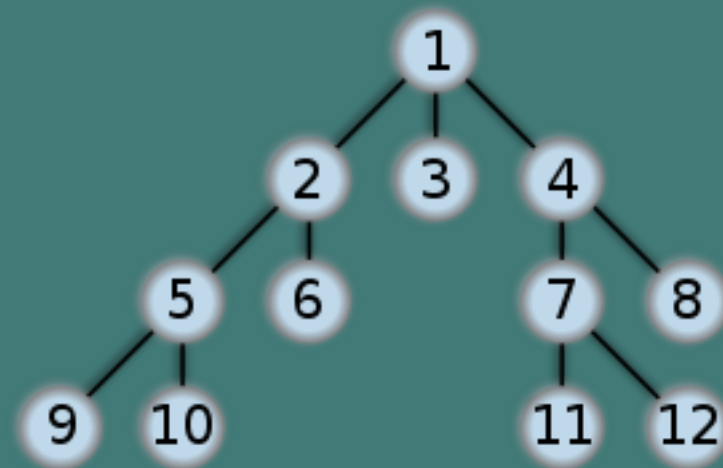
WHAT IS A QUEUE?

WHAT IS A QUEUE?

- In computer science, a queue is a particular kind of abstract data type or collection in which the entities in the collection are kept in order and the principal (or only) operations on the collection are the addition of entities to the rear terminal position, known as enqueue, and removal of entities from the front terminal position, known as dequeue. This makes the queue a First-In-First-Out (FIFO) data structure.

WHAT ARE QUEUES USED FOR?

- Any time you have several actions that need to be executed and only a number of threads of execution (i.e. a bottleneck)

- Printer Spoolers

- Data Buffering (e.g. video streaming)

- Event ticketing

- Checkouts

- Breadth-First Search

HOW TO IMPLEMENT A QUEUE IN RUBY?

```ruby
class Queue
  def initialize
    @list = Array.new
  end

  def enqueue(item)
    @list.push(item)
  end

  def dequeue
    @list.shift
  end

  def peek
    @list.first
  end
end
```

USEFUL RESOURCES

- **Princeton CS: Intro to Programming in Java**

  - *http://introcs.cs.princeton.edu/java/43stack/*

- Carnegie Mellon CS: Intro to Data Structures

  - *https://www.cs.cmu.edu/~adamchik/15-121/lectures/Stacks%20and%20Queues/Stacks%20and%20Queues.html*

- Ruby Monk

  - *https://rubymonk.com/learning/books/4-ruby-primer-ascent/chapters/33-advanced-arrays/lessons/86-stacks-and-queues*

- Propositum: A Ruby Gem by DBC Alum John Akers

  - *https://github.com/johnakers/propositum*

fin