Agenda

Intros:

    WWCode @ Code Fellows PDX

    Link to slides:
    https://github.com/wwcodeportland/study-nights/tree/master/algorithms

Data Structure Summary:

    Queues - How and Why

Lab Time:

    Pair Programming + 3 Queue Algorithms

Algorithms Study Night

WOMEN WHO CODE® PORTLAND

Code Fellows PDX

Leadership Team

Caterina
Director

Richa
Skills Development Lead

Shiyuan
Design Lead

Tricia
DevOps Lead

Alia
Algorithms Lead

Shae
React Lead

Sabina
Events Lead

Sarah Joy
JavaScript Lead

Keeley
Community Lead

Upcoming Events - May

- DevOps Study Night: DevOps and Command Line @ Jive Software
  - Wed, May 3rd, 6 PM
- Workshop: The Roots of Cybersecurity- Introduction to Binary Exploitation @ Galois Inc
  - Sun, May 7th, 10 AM
- Networking Night @ Second Story - Thu, May 11th, 6 PM
- The Circle: Group Viewing @ Living Room Theaters (Downtown)
  - Sat, May 13th, 6 PM
- Design + Product Study Night @ New Relic
  - Tue, May 16th, 5:30 PM
- Workshop: Intro to Git/GitHub, Part 2 @ Mozilla - Wed, May 17th, 5:45 PM
- Roll Call: PyCon 2017 @ Oregon Convention Center - Fri, May 19th, 5:45 PM

{short} Code of Conduct

Women Who Code (WWCode) is dedicated to providing an empowering experience for everyone who participates in or supports our community, regardless of gender, gender identity and expression, sexual orientation, ability, physical appearance, body size, race, ethnicity, age, religion, socioeconomic status, caste, or creed. Our events are intended to inspire women to excel in technology careers, and anyone who is there for this purpose is welcome. Because we value the safety and security of our members and strive to have an inclusive community, we do not tolerate harassment of members or event participants in any form. Our **Code of Conduct** applies to all events run by Women Who Code, Inc. If you would like to report an incident or contact our leadership team, please submit an **incident report form**.

Resources

WWCode @ Meetup.com

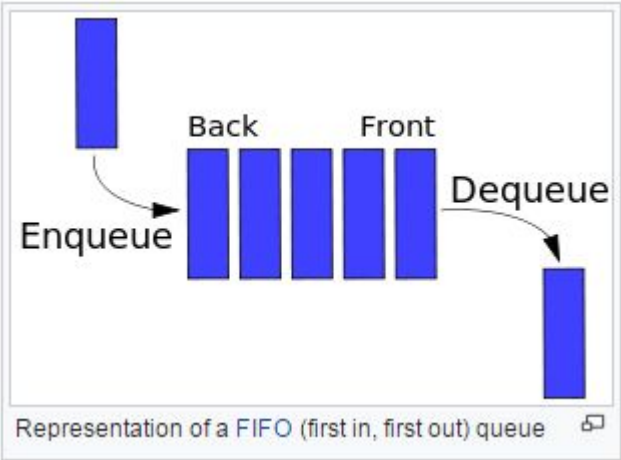
WWCode @ Slack

WWCode @ Github

Big-O CheatSheet

Basics of Queues

- First In First Out(FIFO) structure
- Most common functions include
  - Enqueue - add something to the end of the queue
  - Dequeue - remove the first element in the queue
  - Peek - look at the first element in the queue
  - isEmpty - check if the queue is empty



Representation of a FIFO (first in, first out) queue

Basic Queue Implementation

```
1.  class Queue {
2.      Node first, last;
3.      void enqueue(Object item) {
4.          if (!first) {
5.              back = new Node(item);
6.              first = back;
7.          } else {
8.              back.next = new Node(item);
9.              back = back.next;
10.         }
11.     }
12.     Node dequeue(Node n) {
13.         if (front != null) {
14.             Object item = front.data;
15.             front = front.next;
16.             return item;
17.         }
18.         return null;
19.     }
20. }
```

Additional Information

Javascript Array unshift/pop

Javascript queue implementation

Java Queue Class

C++ Queue Class

Python Queue Module

Ruby Queue Class

C# Queue Class

You can find different implementations of Queues on the Wikipedia page Queue_(abstract_data_type)

3 Queue Algorithms

1	One Task Minion	2	Ping Pong Queue	3	Priority Queue
• Problem Statement		• TopCoder		• TopCoder	