# Archit Mishra

*github.com/armishra · linkedin.com/in/armishra97 · archit72@gmail.com · 510-449-9633*

## Education

2019 - 2020 **M.S. Computer Science**, UC San Diego.
- Accelerated 5 year BS/MS.
- Expected Graduation: June 2020

2015 - 2019 **B.S. Computer Science**, UC San Diego, GPA: 3.78.
- Expected Completion: March 2019
- Honors/Awards: Tau Beta Pi Engineering Honors Society, Provost Honors

## Experience

Jun-Sep 2018 **Facebook**, Software Engineering Intern - Wormhole Pub/Sub Team.
- Reduced subscriber host failure domain by up to 3 times by creating a thrift server API to narrow failure metrics to single subscriptions and act on a subscription level.
- Improved subscriber SLA guarantee by up to 20% by implementing stuck subscriber thread detection.
- Deployed framework for remote procedure call on subscriber hosts to global data centers.

Jun-Sep 2017 **Facebook**, Software Engineering Intern - Database Engineering Team.
- Automated a previously manual process of recovering auto increment values by implementing crash safety into MyRocks server.
- Improved deadlocked transaction handling by building transaction deadlock detection and tracking in RocksDB and MyRocks.

Jun-Aug 2016 **Excelfore**, Software Development Intern.
- Integrated vehicle IOT services with a low latency video camera stream.
- Sped up face detection algorithm by 200% by implementing a prediction algorithm on a 30 MB/s video streaming pipeline.
- Improved lane tracking application by minimizing calculations done per lane candidate leading to a 70% speed up.

2015-Present **UCSD CSE Department**, CS Tutor.
- Undergraduate Tutor for:
  - Computer Architecture [CSE 141]
  - Intro & Advanced Data Structures [CSE 12, 100]
  - Software Engineering [CSE 110]
- Most recently, assisted students in course on MIPS ISA, CPU microarchitecture designs, cache hierarchy, and memory management.

## Projects

Fall 2017 **Openflow Mininet QOS hack**, *C, C++*.
- Implemented token bucket rate limiting into a simple IP forwarding router built on openflow software defined network architecture running on Stanford's mininet emulator.
- Designed a bandwidth allocation algorithm in order to provide differentiated service handling for network packets based on Quality of Service.
- Constructed config based firewall to filter packets by CIDR addresses.

Winter 2016 **Distributed File System**, *Java*.
- Implemented distributed systems mechanisms such as replication and leadership election.
- Designed algorithm to chunk files in parallel and send data over TCP sockets to slaves intended to replicate file chunks.
- Optimized read latency by implementing an in-memory cache for hot chunks.

## Skills/Languages

**Skills**.
- Distributed Systems
- Databases
- Networks
- Storage

**Programming Languages**.
- C/C++
- Python
- Java
- SQL