

YANG DU

1659 Drew Circle APT 1020, Davis, CA 95618
530-746-1627 yondu@ucdavis.edu github.com/duyang0627

EDUCATION

Master of Computer Science	University of California, Davis	Sep. 2014 - Jun. 2016
Bachelor of Computer Science	Beihang University	Sep. 2009 - Jun. 2013

SKILLS

Languages	C/C++, Java, Python, C#, Ocaml, Bash, SQL
Tools	Hadoop, Spark, Qt, Eclipse, Visual Studio, SVN, Git, Linux, Latex

EXPERIENCE

Software Engineer Intern	Cablevision, San Francisco	Jun. – Aug. 2015
Build data pipeline to process real-time data		
Teaching Assistant	University of California, Davis	Sep. 2014 - Present
Classes: Data Structure(ECS 60), Operating Systems(ECS 150);		

PROJECTS

Real-time Data Pipeline for Business Intelligence	Java, SQL	Jun. – Aug. 2015
❑ Technologies explored: Hadoop, Spark, Kafka, Hive, HBase, Redshift		
❑ Collect data and produce to Kafka server		
❑ Consume data from Kafka and process using Spark		
❑ Write processed data into AWS Redshift and MySQL		
❑ Build data dashboard using Tableau		
Research on KLEE - a white box test engine	C++	Mar. 2015
❑ Read papers about test automation, symbolic execution, random testing, fuzz testing, LLVM		
❑ Modified KLEE to support random branch scheduling		
Research on Test Automation	C++/Qt	Jun. 2012 – Jun. 2013
❑ Designed and implemented the test control system of an automated test software that supports editing of test cases, test report generation, and execution of test cases		
❑ Improved the Event Flow Model for GUI testing to adapt to distributed software system		
❑ Implemented automatic test case generation using the improved model		
Operating System Project	C	Apr. – May. 2012
❑ Used Linux system calls extensively		
❑ Including miniature shell, process scheduler and disk management(FAT32)		
“The Harvest” Android Game	Java	Jan. 2012
❑ Initiated game idea and UI design		
❑ Implemented multi-threading to support touch while drawing moving objects		
Pascal Compiler--Compiler Theory Project	C	Nov. – Dec. 2011
❑ Build compiler for a subset of Pascal including functions and procedures		
❑ Convert program to pseudo machine code in one pass using recursive-descent analysis		
❑ Implemented several code optimizations		
“Boxman” - desktop puzzle game	Java	May. – Nov. 2011
❑ Designed and implemented the game UI		
❑ Implemented the gameplay logic with additional features like automatic solving, save and load, map editing and network colloration mode		