Yunfan Wang

yunfanw@andrew.cmu.edu ◇ (412) 209-9349 ◇ www.linkedin.com/in/Yunfan-Wang

EDUCATION

Carnegie Mellon University, School of Computer Science — *Pittsburgh, PA*
M.S. in Information Technology, Privacy Engineering — *Aug. 2016 - Dec. 2017*
GPA: 3.68/4.00

San Jose State University, College of Engineering — *San Jose, CA*
Exchange Program in Computer Engineering — *Aug. 2015 - May 2016*
GPA: 3.875/4.00

Beijing University of Technology, College of Computer Science — *Beijing, China*
B.S. in Computer Science and Technology — *Sep. 2012 - June 2016*
GPA: 3.89/4.00

ACADEMIC PROJECTS

Android Privacy Firewall — *Fall 2016*
Monitor & Filter Network Flow Without Root Permission

- Utilized VpnService in Android APIs to create a self-to-self tunnel, relayed all the network flow through this tunnel before sending out to the Internet and then filtered the network flow in the tunnel without root permission
- Traced Registered Institution of each IP address in American Registry for Internet Numbers's Database (ARIN Whois Database) to help users find the recipient of their released information

Bomberman, a Strategic and Maze-based Online Game — *Spring 2015*
Asynchronous Socket Server & Multi-Threading Game Logic Client Programming

- Utilized C++ and Windows API, including Sockets, Direct2D and Open Database Connectivity (ODBC), to develop a Client-Server Architecture multiplayer online game
- Implemented I/O Completion Ports to efficiently process multiple asynchronous I/O requests from different players; the CPU usage was less than 5% with more than 30,000 connections under Core i7-2600 CPU
- Applied the Frame Locking Technique to synchronize the behavior of each player and distributed the same Pseudo-Random Number Generator seeds to ensure each player had the same random events

C-to-Assembly Compiler — *Spring 2015*
The Declaration, Assignment, Basic Operators and Control Flow parts of Compiler

- Utilized Lex and mechanism of Finite State Automata to generate a lexical analyzer, which matched the desired regular expressions in C source code and output the intermediate file according to the specification given by Yacc
- Applied Yacc to produce a parser to output the Assembly code based on grammar rules

MIPS CPU in a Xilinx FPGA — *Fall 2014*
Single-Cycle, Multi-Cycle and Pipeline Implementation, Interrupt and Exception Handling

- Implemented a MIPS CPU with 50 instructions, pipeline, interrupt and exception using Verilog HDL
- Developed a timer using this MIPS CPU and a Seven-Segment LED; the timer ran successfully after downloading all the codes into a Xilinx FPGA with the ISE

SKILLS

Computer Languages	C++, C, Java, Swift, Python, Perl, Verilog HDL, Assembly
Protocols & APIs	STL, Direct2D, WIN32 API, Socket, I/O Complement Port
Web	J2EE (JDBC, EJB, RMI, JSP, Servlet), HTML, JavaScript, CSS
Databases	Microsoft SQL Server, JDBC, MySQL
Methods & Tools	Design Pattern, UML, Git, Visio, LaTeX

AWARDS

ACM-International Collegiate Programming Contest(ACM-ICPC):

- ACM-ICPC Asia Regional Contest Shanghai Site 2014, **Bronze Medal** — *Dec. 2014*
- ACM-ICPC Asia Regional Contest Anshan Site 2014, **Bronze Medal** — *Oct. 2014*
- ACM-ICPC Asia Regional Contest Changchun Site 2013, **Bronze Medal** — *Dec. 2013*

The IEEEXtreme Programming Competition 7.0, **Global Ranked 18th** — *Oct. 2013*