Kyle Fung

University of Waterloo Computer Science 3B
Phone: (647)-770-5100 Email: kjfung@uwaterloo.ca
Website: kylefung.github.io

TECHNICAL SKILLS

Languages: C/C++, Java, CUDA C, HLSL, GLSL
Tools: Bash, GDB, Visual Studio, APITrace, Git, Mercurial
API/Platforms: OpenGL, DirectX, CUDA
Domains: Computer graphics, Compilers

EDUCATION

Program – Bachelors of Computer Science (class of 2018) *September 2013 – present*

- Cumulative average of **89% (3.98 GPA)** with an average of **91.6%** in Computer Science courses
- Relevant courses: Computational Linear Algebra (CS475), Numerical Computation (CS370), and Operating Systems (CS350)

Research – Undergraduate research assistant *January 2016 – present*

- Studied computational fluid dynamics under Dr. Christopher Batty during the school term
- Learned state of the art solutions and techniques through reading academic papers and internet research
- Engineered and implemented a 2D interactive real-time fluid simulator

WORK EXPERIENCE

Mozilla Corporation – Graphics engineering intern *May 2015 – August 2015*

- Regularly debugged, wrote and maintained C++ over a massive code base
- Fixed conformance issues in the behavior of Firefox's WebGL implementation
- Added WARP device support for WebGL using ANGLE
- Started upgrade of test infrastructure to use WebGL conformance test suite version 1.0.3
- Diagnosed and fixed rendering issues with Firefox on Windows

TransGaming Inc. – Graphics and portability developer *August 2014 - December 2014*

- Debugged large C++ applications to diagnose rendering issues
- Set up more than 1000 rendering tests using the OpenGL ES2 conformance suite, rendered using ANGLE
- Wrote over 70 HLSL shader programs to test sanity of an HLSL to GLSL compiler

IBM Canada – Infrastructure developer *January 2014 - April 2014*

- Maintained an automated testing system over a network of 50 servers for IBM's JIT compiler team

PERSONAL PROJECTS

WasteEngine (github.com/KyleFung/wasteEngine)

- A toy rendering engine using OpenGL, written in C++, supporting lighting and custom shaders
- Implemented basic model loading with Assimp and texture loading with ImageMagick

FluidCanvas (github.com/KyleFung/fluidCanvas)

- An interactive liquid and smoke simulator based on numerical techniques, written in Javascript

Recursive Ray Tracer (github.com/KyleFung/RayTracer)

- An offline ray tracing renderer to accurately display 3D scenes using Blinn-Phong lighting, in C++

Voxel Renderer (github.com/KyleFung/smokeBox)

- A precursor to an implementation of the marching cubes algorithm, written in C++, CUDA C, and OpenGL