

# Tristan Amini

(416) 801-5288 | tsamini@uwaterloo.ca

## EDUCATION

**University of Waterloo**  
SYSTEMS DESIGN ENGINEERING  
Class of 2018

## LINKS

Github: [github.com/Tamini](https://github.com/Tamini)  
LinkedIn: [linkedin.com/in/tamini](https://linkedin.com/in/tamini)

## PROGRAMMING

### Language | Proficiency

|                                 |      |
|---------------------------------|------|
| Java                            | ++++ |
| C++                             | ++++ |
| Go                              | +++  |
| L <sup>A</sup> T <sub>E</sub> X | +++  |
| OCaml                           | +++  |
| Scala                           | +++  |
| SQL                             | +++  |
| Python                          | +++  |
| MATLAB                          | +++  |
| R                               | +++  |
| Assembly                        | ++   |
| Chef                            | ++   |
| Ruby                            | ++   |

### Tool | Proficiency

|            |      |
|------------|------|
| Git        | ++++ |
| Jira       | ++++ |
| gdb        | ++++ |
| Jenkins CI | ++++ |
| WinDebug   | +++  |
| Perforce   | +++  |
| Maven      | +++  |
| Android    | +++  |

## PAST EMPLOYERS

### Interaset

"...he demonstrated an exceptional ability to very quickly become productive with new technologies."

- Michael Iles | Dev Lead

### IBM

"...his innovative solution to managing exceptions and frame descriptors was the key to our finishing that project..."

- Bill O'Farrell | Developer

This resume is written in L<sup>A</sup>T<sub>E</sub>X!

Find the source at:

[github.com/Tamini/Resume](https://github.com/Tamini/Resume)

## WORK EXPERIENCE

### AMD | DRIVER DEVELOPER

January 2016 - April 2016 | Markham, ON

- Fixed bugs in the AMD Kernel Mode Driver for Windows 7, 8.1 and 10.
- Led the team that created a C++ program to simulate attacks on the AMD display driver. Fixed all identified security vulnerabilities leading to a more secure and reliable driver for AMD's users.

### IBM | OPEN SOURCE COMPILER DEVELOPER

May 2015 - August 2015 | Markham, ON

- Part of the two man team that ported the OCaml compiler to the Z Architecture. Required reading and writing x64, PPC and Z Assembly.
- Wrote a relocation type for the Go linker to work with Z.
- Worked on the Go Assembler for Z.

### INTERSET | SOFTWARE ENGINEER

September 2014 - December 2014 | Ottawa, ON

- Wrote bash script to allow for automatic OVA creation for customer deployments of software bundle.
- Created Scala application to read and anonymize millions of data entries.
- Maintained and set up a Hadoop/HBase/Phoenix/Spark install on AWS.
- Built web pages for use internally and by customers.

### POLAR | DEVOPS

January 2014 - April 2014 | Toronto, ON

- Set up easily recreatable and destroyable virtual machines using Chef.
- Created MySQL dump scripts in Python and SQL that run daily using cron.
- Wrote an IRC bot in Python to help coordinate users and add useful features to the IRC chat.

## PERSONAL PROJECTS

### SPARK | THIRD-PERSON HACK AND SLASH GAME

March 2016 - Present | Still in development!

Creating a third-person PC game in Unreal Engine 4 using Blueprints and C++. Currently have enemy NPCs, basic animations and basic combat. Will require working with game AI, animating, modelling, texturing and programming.

### MARKOV CHAIN | GENERATE TEXT BASED ON INPUT

July 2015 | [github.com/Tamini/markov-test](https://github.com/Tamini/markov-test)

Simple Java program that implemented a Markov Chain. Input text could be read from a text file and stored in a HashMap. Text would be generated based on probability of a specific word coming after another word. Given enough sample text as input, it would generate new text based on the learning acquired.

### ANTICHESS | CHESS GAME

2012 - 2013 | [github.com/Tamini/AntiChess](https://github.com/Tamini/AntiChess)

Created a chess game in Java with a custom AI in Java. AI implemented alpha-beta pruning and depth-first search to find the best moves. Each piece was assigned a point value and the AI determined the best move by looking at possible future board states and determining the best outcome.