Tyler King

tylerking841@gmail.com | github.com/tylerking841 | linkedin.com/in/tylerkiingg

Technical Skills

- Languages: C, C++, C#, Java, JavaScript, HTML, CSS, PHP, SQL
- Frameworks: React, .Net, ASP.NET, JUnit
- Development Tools: Git, Azure, VS Code, Visual Studio, Eclipse, NetBeans, IntelliJ, Postman, Wireshark

Education

Algonquin College, Ottawa, ON September 2019 - December 2022

Ontario College Advanced Diploma

Computer Engineering Technology - Computing Science

- Relevant Courses: Object-Oriented Programming, Data Structures, C Language, .NET Enterprise Appl. Dev, Operating Systems, Web Enterprise Applications (Restful API), Software Design and Testing, Compilers, Introduction to Database

Relevant Experience

Fast Food Ordering Website

- Online food ordering web application developed in React with OAuth user login
- Heavily contributed on testing and documentation in team of 6 in an Agile development environment
- Liaised consistently with real stakeholders to incorporate feedback into end product

Restful API Sprite Simulator

- Designed and developed a RESTful API using Java Enterprise Edition (Java EE) technologies such as JAX-RS, EJB, and JPA.
- Integrated the API with Heidi SQL to persist and retrieve data.
- Sprite location updated real time in database, can be viewed and edited on HTML page as JSON values.
- Developed and tested on local Glassfish server

Custom C Language Compiler

- Designed and developed a custom C language compiler
- Implemented a custom lexer to recognize and tokenize the unique language syntax, using regular expressions and state machines.
- Designed and implemented a custom parser to parse and validate the language grammar, using recursive descent parsing and LL(k) parsing techniques.
- Integrated the buffer and scanner into the compiler pipeline, ensuring seamless and efficient processing of the source code.

Java 3D Lithophane Generator

- Utilized object-oriented programming concepts, such as inheritance and polymorphism, to build a modular and scalable application.
- Implemented mathematical algorithms to generate fractal patterns, using 2D arrays to store and manipulate the data.
- Utilized Java unit testing frameworks, such as JUnit, to write and run automated tests, ensuring code quality and stability.