PATRICK COPP

github.com/patrickcopp
905-923-8958 ⋄ pcopp@uoguelph.ca

SKILLS

Languages

· C(ANSI & C99), Python, OracleSQL/ MySQL, VBA, Java, C++, JavaScript, HTML, Assembly, R

Tools/ Skills

· PL/SQL, Regex, AWS EC2/S3, ETL, Git, Node.js, Bash, Swing, Microsoft Office

EDUCATION

University of Guelph September 2017 - April 2022

· Bachelor of Computing, Computer Science (Co-op) Minor in Business Economics
· Nominated for **Ian Pavlinic Memorial Award (Co-op Student of the Year)**
· 91% achieved in CIS 2750 (Software Systems Development and Integration)
· Major GPA of 79%
· Dean's Honour List (average above 80%) achieved Fall 2018 semester

WORK EXPERIENCE

ERIS - Environmental Risk Information Services May 2019 - December 2019
Data Technician, Data Analyst

· Headed project for collection of more than 150 years of records for over **500,000 wells** using **Selenium, urllib3, and PL/SQL**
· Independently created, tested, and ran programs to capture data from 6 separate websites; collected more than **80GB** of City Directories (images, .pdf files) using **BeautifulSoup4** and urllib3
· Consulted on the development of an in-house implementation of **OCR machine learning** to read 10,000s of pages of City Directories
· Wrote queries and **PL/SQL** procedures to perform analysis, normalization, and **ETL** (Extract, Transform, Load) on dozens of publicly available datasets in a variety of different formats (.csv, shapefile, .xlsx, .pdf) in an Oracle environment
· Worked with ESRI software to geolocate U.S. addresses in a variety of formats and assess results; wrote **Regex** statements to clean Canadian addresses in preparation for geocoding

PROJECTS

iCal Parsing Web App January 2019 - April 2019

· Created full-stack web application to parse and store iCal files (Outlook, Apple Calendar, Google Calendar)
· Offered back-end service of **C** to parse, and **SQL** to store calendars
· Front-end service used **HTML, JavaScript, and CSS**

Tic-Tac-Toe January 2018

· Created a perfect Tic-Tac-Toe program using ANSI **C** that is unbeatable
· Used **Git** to track progress of the project
· Utilized 9 binary files to store all possible conditions of board state

Slide Puzzle June 2017

· Created a slide puzzle game that automatically created a solvable board state for players using **Java**
· Offered 4 levels of difficulty and board size as well as differing high score text files for each level
· Utilized **Slick2D** to create a graphical interface for the game