Michael Pham

ktm-p.github.io

Email: ktmpham@berkeley.edu
Mobile: (916)-968-0563

Education

- **River City High School** West Sacramento, CA
  Salutatorian *Mar 2019 – Jun 2022*
  - GPA : 4.00
- **University of California, Berkeley** Berkeley, CA
  B.A. in Computer Science and Mathematics; Minor in Data Science *Aug 2022 – Present*
  - GPA : 3.86
  - Member of Upsilon Pi Epsilon Honor Society

Projects

- **Audio Analyzer** | Java, Processing
  - Displays different representations of the audio, giving user information on chosen mp3 file.
  - Includes audio waveform and polar graph representation of audio.
  - Implemented a Discrete Fourier Transform algorithm. Smoothed the RDFT representation of the audio as well.
  - Includes a beat detection feature by observing the audio's level and seeing if there's a marginal difference.
- **Berkeley Admissions Visualization** | Python, Matplotlib, NumPy, Pandas, Plotly, RegEx, Seaborn
  - Compiled data on Berkeley's Californian public school admissions, and created visualizations for it.
  - Worked with datasets from multiple different sources with different formatting.
  - Utilized Pandas and RegEx in order to filter down datasets, regularize them, and merge them.
  - Used Seaborn and Matplotlib to visualize certain statistics (race, gender, etc.) with boxplots, bar charts, etc.
  - Created scattermaps and choropleth maps using Plotly to aid in seeing areas with higher admissions.
- **Build Your Own World** | Java
  - An interactive maze exploration survival game featuring enemies.
  - Implemented a pseudo-random world generation system via Prim's Algorithm.
  - Utilized BFS to implement a smooth lighting system, and obstructing player vision around obstacles.
  - Enemies equipped with pathfinding AI implemented with A*-search Algorithm.
  - Programmed saving functionalities through serialization.
- **Optimizing Convolutions** | C, OpenMP, OpenMPI, SIMD
  - Implemented a naïve 2D Convolution algorithm, then further optimized it. Achieved around a x50 speedup.
  - Switched to working directly with pointers rather than with array accesses in order to speed up performance.
  - Reordered order of instructions in order to optimize cache usage.
  - Stored commonly-used variables in registers in order to better execution time.
  - Used OpenMP to implement parallel programming to enhance runtime.
  - Vectorized operations using SIMD instructions, and implemented extensive loop unrolling as well.
- **Spam Classifier** | Python, Matplotlib, NumPy, Pandas, RegEx, scikit-learn, Seaborn
  - Created a spam email filter using a Logistic Regression model. Achieved an accuracy of 99.2% on given test data.
  - Visualized relationships between features, and distribution of certain phrases with Matplotlib and Seaborn.
  - Cleaned and filtered data using Pandas.
  - Extracted key terms, phrases, and character sequences with RegEx.
  - Fine-tuned hyperparameters by cross-validation with GridSearchCV.

Technical Skills

- **Programming Languages**: C, CSS, Golang, HTML, Java, Javascript, MATLAB, Python, R, RISC-V, Scheme, SQL
- **Frameworks/Libraries**: Matplotlib, Numpy, OpenMP, OpenMPI, Pandas, Plotly, Processing, PyTorch, scikit-learn, Seaborn, TensorFlow
- **Tools**: Docker, gdb, git, Logism, Valgrind
- **Mathematics**: Abstract Algebra, Discrete Mathematics, Linear Algebra, Logic, Numerical Analysis, Real Analysis