Software Engineering

In this class I learned a lot about the software engineering process that comes with developing a product. The software life cycle can vary widely between companies and which methodology the company uses such as iterative (Agile) or waterfall. I learned about requirements, formal specification and validation, techniques for software design and testing, cost estimation models, and issues in software quality assurance and software maintenance. This allowed me to take into consideration the overall process (requirements to maintenance) when beginning on a new project like my student organization student app.

Operating Systems

In this class I learned more about how operating system manages resources. This included learning about their design, implementation, and usage of operating systems. In addition, I further learned about main memory management, virtual memory, I/O and device drivers, file systems, secondary storage management, critical sections, deadlocks, operating system algorithms, and more. This allowed me to create more resourceful and faster programs using selectively threads/processes.

Database Systems

In this class I learned how to design and implement a database management system. Topics included data models such as learning the ER model and the EER model. Furthermore, I learned data description languages such as DDL, DML, and more. Also learned about data normalization using the first normal form, second normal form, third normal form, Boyce-Codd Normal Form, and even the fourth normal form. Other areas that I learned was query facilities, file organization, index organization, file security, data integrities, and reliability. This allowed me to optimize my data storage for my projects that use Firebase.

Mathematical Foundation

In this class I learned a lot about Boolean logic, first-order logic, models of first-order logic, program verification, completeness theorem, regular expressions, regular sets, finite-state machines, graph theory, graph algorithms, state charts, and petri nets and their in software engineering. This class has helped me think differently when approaching algorithms, creating algorithms, and preceding with any Boolean logic.

Computer Architecture

In this class I learned a lot about the operating system, the CPU, and instruction set architecture of the hardware. I learned how the ISA is used to do basic computations and reads instructions. In addition, I learned about the MIPS assembly language, the translation between MIPS and C, performance calculation, processor data paths, pipelining, memory hierarchy, and more. This class has allowed me to further understand the bare bones of computers and ultimately coding as it showed me to avoid redundancy and provide simple optimizations in my programs.

Probability and Statistics

In this class I learned about how probability and statistics can be used in computer science and software engineering. The things that I learned were the axiomatic probability theory, independence, conditional probability, discrete and continuous random variables, special distributions of importance to CS/SE, simulation of random variables, Monte Carlo methods, central limit theorem, basic statistical inference, parameter estimation, hypothesis testing, linear regression, and more. Although I have yet to fully use my skills gained in this class, I hope to use it later in speech recognition, data mining, compression, and more.

Data Structures and Introduction to Algorithmic

In this class, I learned many data structures and algorithms that include stacks, queues, tree (including a variety of trees like B-trees), heaps, hash maps, hashing, disjoint sets and graphs, sorting algorithms, searching algorithms and more. In addition, I learned how to optimize algorithms, detect bottlenecks, and analyze algorithms using time complexity and Big-O notation. This has helped me provide more efficient code in my projects and further understand algorithms and data structures.

Programming in Unix

In this class I learned how to program in C and Python using a Unix environment. I learned basic Unix concepts, file input and output, implementation of strings, dynamic memory allocation/management, and learned how techniques utilizing procedural and object-oriented programming in a Unix environment. This gave me an introduction to Python and gave me a further understanding of C, Unix, client and server modal, and SSH.

Computer Science II

In this class I learned how to program in Java. During this class I was able to advance my programming techniques and understand data structures and algorithms in Java. This includes recursion, data structures (stacks, queues, linked lists, hash tables, trees, graphs, etc.) and showed me some algorithmic analysis. This class allowed me to learn Java and prepare me for more advance classes.

Discrete Mathematics

In this class I learned the beginnings of Boolean operations, Boolean logic, proof methods, recurrence relations, sets, relations, functions, elementary number theory, and more. This class paved the beginning for me in terms of understanding the logic behind coding, proofs, algorithms, and algorithmic analysis.

Computer Science I

In this class I learned how to further my understanding of the programming language C++. This showed me how to do object-oriented programming in C++ with a focus on control structures and structured data types. In addition, classes were a strong emphasis with the inclusion of basic analysis of algorithms, searching and sorting techniques, and an introduction to software engineering. This class showed me for the first time how to code using objects and classes.

Linear Algebra

In this class I learned a lot about linear algebra that includes models for application of the concepts of vector algebra, finite dimensional vector spaces, geometric significance, Gaussian elimination, matrix inversion, matrices determinants, linear transformation, quadratic forms, eigenvalues, eigenvectors, and more. Although haven’t directly applied what I learned so far, I am very excited to apply soon, once I start my machine learning and artificial intelligence courses.

Programming Fundamentals

This class was my very first course in programming. I learned a lot in this course. This course taught me techniques used in programming, coding styles, and the programming language C++. In addition, it helped me learn data types, structures, functions, arrays, how to run programs, how to test programs, and how to debug programs. This class was extremely influential in me as it showed what is coding and how to code.