

## **Autumn Quarter 2019: Freshman Year (Computer Science)**

### **CSE 311 Foundations of Computing I (Kevin Zatloukal):**

Examines fundamentals of logic, set theory, induction, and algebraic structures with applications to computing; finite state machines; and limits of computability.

### **CSE 351 Hardware/Software Interface (Justin Hsia):**

Examines key computational abstraction levels below modern high-level languages; number representation, assembly language, introduction to C, memory management, the operating-system process model, high-level machine architecture including the memory hierarchy, and how high-level languages are implemented.

### **CSE 391 System and Software Tools (Zorah Fung):**

Introduction to tools commonly used in software development. Topics include using a command-line interface, writing scripts for file and string manipulation, managing user permissions, manipulating text with regular expressions, using build-management tools, and using version-control systems.

## **Winter Quarter 2019: Freshman Year (Computer Science)**

### **CSE 312 Foundation of Computing II (Martin Tompa):**

Examines fundamentals of enumeration and discrete probability; applications of randomness to computing; polynomial-time versus NP; and NP-completeness.

### **CSE 332 Data Structures and Parallelism (Kevin Lin):**

Covers abstract data types and structures including dictionaries, balanced trees, hash tables, priority queues, and graphs; sorting; asymptotic analysis; fundamental graph algorithms including graph search, shortest path, and minimum spanning trees; concurrency and synchronization; and parallelism.

### **CSE 492 Career Seminar (Kimberly Nguyen):**

Covers basics of the interview process, behavioral interviews, coding interviews, how to get interviews and so on.

## **Spring Quarter 2020: Freshman Year (Computer Science)**

### **CSE 331 Software Design and Implementations (Kevin Zatloukal):**

Explores concepts and techniques for design and construction of reliable and maintainable software systems in modern high-level languages; program structure and design; program-correctness approaches, including testing; and event-driven programming (e.g., graphical user interface). Includes substantial project and software-team experience.

### **CSE 473 Introduction to Artificial Intelligence (Steven Tanimoto):**

This course begins with a discussion of what "artificial intelligence" means and how it can be useful. Next is a grounding in state-space search, problem formulation and solving. We then consider state-space search in the context of game playing, which then leads to alpha-beta pruning, expectimax search and the modeling of uncertainty. After that, we

focus primarily on machine learning, starting with methods for reasoning under uncertainty. We cover reinforcement learning in Markov Decision Processes, perceptron learning, and deep learning.

**CSE 421 Introduction to Algorithms (Shayan Oveis Gharan):**

Techniques for design of efficient algorithms. Methods for showing lower bounds on computational complexity. Particular algorithms for sorting, searching, set manipulation, arithmetic, graph problems, pattern matching.

**CSE 131 Science and Art of Digital Photography (Bruce Hemingway):**

Covers the fundamentals of digital photography, including computational imaging; the elements of photographic composition and design; and the future of internet-enabled photography.

**Summer Quarter 2020: Freshman Year (Computer Science)**

**CSE 344 Introduction to Data Management**

**CSE 333 Systems Programming**