

Teaching Evaluations

Link to repository: github.com/Parsa-Rajabi/academic

A repository to share my teaching evaluations as a teaching assistant and instructor positions.

University of Fraser Valley (UFV)

List of courses as an Instructor:

- CIS 100: Introduction to Computers and Digital Technologies
- CIS 145: Web Publishing
- CIS 385: Project Management

CIS 100 - Introduction to Computers and Digital Technologies [3]

Hands-on approach to learning the basics of operating systems, file management, knowledge management, cloud computing, and communications technology. Learn software skills relevant to student success in other courses, such as word processing, presentations, spreadsheets, and graphics.

CIS 145 - Web Publishing [3]

In this course, students will design, organize, publish, and maintain a basic website using HTML. They will also learn scripting, how to apply multimedia plug-ins, and cascading style sheets. This course is intended for students with no prior knowledge of HTML.

CIS 385 - Project Management [3]

This course is designed to complement existing information technology knowledge, skills, and experience with the project management skills required to effectively manage development projects that involve computer hardware, computer software, and telecommunications technology.

Langara College

List of courses as an Instructor:

- CPSC 2350: Software Practices

CPSC 2350 - Software Practices [3]

Students learn how to apply software development best practices to create the right product (validated), done right (verified), and managed right (through responsive and responsible process). They will learn the “what”, “how”, and “why” of agile practices, and the consequences for not following them, while

working through the Software Development Life Cycle (SDLC) on multi-developer projects.

Simon Fraser University (SFU)

List of courses as an instructor:

- CMPT 276: Introduction to Software Engineering

List of courses as a Teaching Assistant:

- CMPT 120: Intro to CS and Programming
- CMPT 363: User Interface Design
- CMPT 767: Visualization

CMPT 120 - Introduction to Computing Science and Programming I [3]

An elementary introduction to computing science and computer programming, suitable for students with little or no programming background. Students will learn fundamental concepts and terminology of computing science, acquire elementary skills for programming in a high-level language, e.g. Python. The students will be exposed to diverse fields within, and applications of computing science.

CMPT 276 - Introduction to Software Engineering [3]

An overview of various techniques used for software development and software project management. Major tasks and phases in modern software development, including requirements, analysis, documentation, design, implementation, testing, and maintenance. Project management issues are also introduced. Students complete a team project using an iterative development process.

CMPT 363 - User Interface Design [3]

This course provides a comprehensive study of user interface design. Topics include: goals and principles of UI design (systems engineering and human factors), historical perspective, current paradigms (widget-based, mental model, graphic design, ergonomics, metaphor, constructivist/iterative approach, and visual languages) and their evaluation, existing tools and packages (dialogue models, event-based systems, prototyping), future paradigms, and the social impact of UI.

CMPT 767 - Visualization [3]

Advanced topics in data visualization. Topics covered may include principles of data representation, data presentation, data interaction, data physicalization, data and visualization literacy, data visualization and diversity, open data, and public personal data visualization.

University of British Columbia (UBC)

List of courses as a Teaching Assistant:

- COSC 101: Digital Citizenship
- COSC 111: Computer Programming I
- COSC 122: Computer Fluency
- COSC 221: Intro to Discrete Structures
- COSC 310: Software Engineering

COSC 101 - Digital Citizenship [3]

Knowledge and skills to navigate the digital society. Digital participation, digital access, skills and utilization. Digital literacy, computer applications, converging technologies, and online resources. This course does not assume students have any Computer Science background

COSC 111 - Computer Programming I [3]

Introduction to the design, implementation, and understanding of computer programs. Topics include problem solving, algorithm design, and data and procedural abstraction, with emphasis on the development of working programs.

COSC 122 - Computer Fluency [3]

Introduction to computer skills (electronic communication, websites, Internet, document editing, programming, data analysis using spreadsheets/databases) and concepts (information representation, abstraction, algorithmic thinking). Course objectives are lifelong productivity and understanding of technology in society.

COSC 221 - Introduction to Discrete Structures [3]

Sets, logic, combinatorics, and graph theory, as applied in computing: sets and propositions, permutations and combinations, graphs and trees, Boolean algebra, algorithms, and applications.