

COMPUTING DISCIPLINES

Computer Game Design and Development (CGDD)

The CGDD program exposes students to the field of computer game design and development, including digital media, human-computer interaction, the history and theory of gaming, game design, 2D and 3D graphics, simulation, modeling, software engineering, artificial intelligence, data structures and algorithms. Current and emerging domains include online games and massively multiplayer games (MMOG), casual games, mobile games, and serious/educational games are explored.

Computer Science (CS)

Computer Science emphasizes the study of computer systems architecture, software development, and data communications. Core technology areas include programming, computer architecture, operating systems, data communication, database systems, and software engineering. These areas are supported by a strong foundation in computing principles such as the design of programming languages, data structures, and operating system principles. Additionally, mathematics concepts are incorporated into many of the major courses.

Information Technology (IT)

Information Technology (IT) in its broadest sense encompasses all aspects of computing technology. IT is concerned with issues related to advocating for users and meeting their needs within an organizational and societal context through the selection, creation, application, integration and administration of computing technologies. IT programs aim to provide IT graduates with the skills and knowledge to take on appropriate professional positions in Information Technology upon graduation and grow into leadership positions or pursue research or graduate studies in the field. Students further focus their studies into certain fields by choosing a concentration in **Enterprise Systems**, **Information Security**, **Health IT**, or **Mobile and Web Development**.

Software Engineering (SWE)

Be a part of the only undergraduate program of its kind at a public university in Georgia. Building on a base of computer science and mathematics, SWE students learn to engineer software systems through the entire life cycle: requirements, design, implementation, testing, integration, and maintenance. Students work on teams that deliver high-quality software products. Coursework is selected from areas such as computer game design and development, user interaction engineering, component-based development, embedded systems, and other state-of-the-art methodologies and practices that reflect the fast paced changes in this dynamic field.