**ADVANCED PROGRAMMING COURSEOUTLINE**

**IAS/Defensive Programming/Programming for Security**

•Secure software development and coding practices (cross-reference SE/Software Construction)

• building security through the software development Life Cycle using Threat modeling and vulnerability identification

• Web and program security Application vulnerabilities and defences and development methods

Examples of proper coding methods input validation and data sanitization errors (Chapter1 and 2)

Program/Application Correctness and vulnerabilities and defences)(chapter3 and 4)

o Buffer overflows

o Integer errors

o SQL injection

o XSS vulnerability

o CSRF

Session management, authentication(chapter 6)

o Single sign-on

o HTTPS and certificates

• Client-side security(chapter 7)

o Cookies security policy

o HTTP security extensions, e.g. HSTS

o Plugins, extensions, and web apps

o Web user tracking

• Server-side security tools, e.g. Web Application Firewalls (WAFs) and fuzzers(chapter 8)

Programming for Robustness and maintainance

• Proper coding and development standards plus choice of type-safe languages

• Race conditions (cross reference SF/Parallelism/Parallel programming and PD/Parallel Architecture/Shared vs. distributed memory and PD/Communication and Coordination/Shared Memory and PD/Parallelism Fundamentals/Programming errors not found in sequential programming)

• Correct handling of exceptions and unexpected behaviors (cross reference SDF/Development

Methods/program correctness)

• Using assertions

• Using regular expressions to handle error plus software bugs

•Code evaluation/code testing

• Correct usage of third-party components/Libraries

• Effectively deploying security updates

• Code reusability

**IAS/Principles of Secure Design as the guidelines security trends**

Least privilege and isolation

Fail-safe defaults

Open design

End-to-end security

Defense in depth (e.g., defensive programming, layered defense)

Security by design