Security-Related Computer Science and BAIS courses at Xavier University, 2021-2022

**CSCI 170 - Computer Science I**
3 Credit Hours
This course is an overview of computer science. Topics include problem solving and algorithms, machine architecture, operating systems, assembly language, higher level programming languages, compilers, limits of computation, networking, applications, and social/ethical issues.
Course Attributes: Mathematical Perspectives

**Outcomes:**
- explain key notions of computer science, particularly algorithm, abstraction, computability, and efficiency in non-technical terms.
- read articles/book about technology or computer science and relate them to the notions of computer science discussed in class.
- explain the basic components of algorithms and use them to describe algorithms for simple tasks, demonstrating algorithmic problem-solving skills. (The environment for the algorithms will vary and students will be expected to demonstrate skills in each environment.)
- design abstractions for a variety of problems.
- analyze some simple algorithms in terms of their efficiency.
- reflect on the ethical issues raised by activities facilitated by technology related to computer science.

**CSCI 180 - Computer Science II**
3 Credit Hours
Program design: advanced syntax of a programming language; dynamic memory; recursion; sorting; searching; stacks; queues; social and ethical issues related to software design and reliability.
Prerequisites: Undergraduate level CSCI170 Minimum grade of D

**Outcomes:**
- Read and analyze text, including: explaining code; using documentation to determine what code does; analyzing problem descriptions to determine the tasks necessary for solution;
- Design solutions to problems, implement them in java, document the solution at a level that allows beginning programmers to understand your solution, and develop test cases to provide evidence your solution is correct;
- Conduct independent explorations into the course material, including development of plans for exploring possible solutions and acting independently on that plan;
- Identify possible extensions to problems you work on, exploratory questions about the topics in the course, and multiple methods for solutions; and
- Analyze ethical questions related to algorithms and software development
CSCI 210 - Machine Org & Assembly Lang
3 Credit Hours
Machine level representation of data, assembly level machine organization, memory system organization and architecture, number representation and errors, assembly language.
Prerequisites: Undergraduate level CSCI170 Minimum grade of D

Outcomes:
• Describe a modern computer architecture.
• Explain how modern architecture impacts performance and energy efficiency by analyzing tradeoffs on a given problem.
• Use the eight great ideas of computer architecture as a way to analyze decisions in machine design.
• Demonstrate basic ability to create and analyze assembly language programs.
• Relate assembly language programs to higher level programming constructs.
• Demonstrate ability to use machine-level representations for several data types

CSCI 220 - Data Structures & Algorithms
3 Credit Hours
Trees, hashing, advanced sorts, numerical algorithms, algorithm analysis, algorithm design and problem-solving strategies.
Prerequisites: Undergraduate level CSCI180 Minimum grade of D
Course Attributes: Quantitative Reason Flag Full

Outcomes:
• Analyze and prove worst case arguments about algorithms and data structures involved in a wide variety of problems,
• Analyze and evaluate problems, describing the pros and cons of various solutions (including an analysis as described in objective 1),
• Describe and implement a variety of key data structures and algorithms, including hash tables, heaps, stacks, queues, binary search trees, depth-first-search, quicksort, shortest path algorithms, and minimum spanning tree algorithms,
• Demonstrate an ability to independently program solutions for real world problems involving the algorithms and data structures covered in the course,
• Analyze the impact of design choices on the end user,
• Express how ethical and social justice issues are integrated into computational algorithms,
• Describe how bias impacts project design choices, and
• Communicate with non-technical individuals about a technical project.
CSCI 240: Systems Security Fundamentals
3 Credit Hours
Introduction to system level security issues. Operating systems, networks, and databases are examined from a design perspective to recognize the security threats in current systems and design principles for secure systems. Includes examination and use of tools for monitoring and administering systems securely.
Prerequisite: CSCI 220

Outcomes:
• Demonstrate an understanding of cyber defense, threats, and security design principles, including analyzing systems to explain design principles needed and violated, and the actions that can be taken to exploit and defend the system.
• Articulate laws and policies, ethical issues on cybersecurity.
• Demonstrate an ability to perform system administration duties for installing, updating, and monitoring a system, including scripting and command-line tools for penetration testing, and network monitoring.
• Understand key concepts of secure networking protocols and hypothesize about their weaknesses.
• Describe security models for operating systems and hypothesize about their weaknesses.
• Describe security models for database design and deployment hypothesize about their weaknesses.

CSCI 250 - Languages & Automata
3 Credit Hours
Finite-state-automata and regular expressions, context-free grammars, pushdown automata. Turing machines, computability and undecidability, complexity classes.
Prerequisites: (Undergraduate level MATH180 Minimum grade of D or Undergraduate level MATH225 Minimum grade of D) and Undergraduate level CSCI170 Minimum grade of D

Outcomes:
• Explain terms and concepts involved in CS theory, both to non-experts and with the preciseness and accuracy that experts would use. Examples of these terms and concepts are: finite automata, non-determinism, context free grammars, regular expression, turing machines, NP-Completeness, language acceptance, and reduction.
• Relate the technical terms and concepts of CS theory to real problems encountered in more applied areas of computer science, (i.e. Be able to answer the "So what?" question.)
• Apply the concepts and tools such as finite automata, grammars, and reductions to solve particular problems.
• Write proofs or provide counterexamples to demonstrate a variety of properties about the concepts studied in the course. This includes both replication, with understanding, of established theorems in this area and development of proofs or counterexamples when you encounter new questions in this area.
CSCI 320 - Operating Systems
3 Credit Hours
Operating system software and hardware design and implementation; tasks and processes; process coordination, synchronization, and scheduling; physical and virtual memory organization; file systems and naming; security and protection.
Prerequisites: Undergraduate level CSCI210 Minimum grade of D and Undergraduate level CSCI220 Minimum grade of C

Outcomes:
- Describe what an operating system is and how it accomplishes its primary tasks.
- Demonstrate how operating systems both implement and support concurrency.
- Compare and contrast the performance implications of various resource sharing algorithms: CPU scheduling, virtual memory/page replacement, disk scheduling, etc.
- Implement all the key components of a simple, virtual memory capable operating system.
- Describe what a file system is and how they are implemented.

CSCI 335 - Databases
3 Credit Hours
Database systems are the software systems used to manage large volumes of data. The principles of database systems, hardware characteristics, file organization and evaluation, data models, database schemas, etc., are studied from both a theoretical and practical viewpoint. The emphasis will be on solving the problems encountered in designing and using a database system, regardless of the underlying hardware and operating system on which the database system will run.
Prerequisites: Undergraduate level CSCI220 Minimum grade of C

Outcomes:
- Model an enterprise in terms of a data model; e.g. schema in the relational data model.
- Describe how DBMSs leverage abstraction to provide data independence.
- Write moderately complex queries in both relational algebra and SQL.
- Compare and contrast the performance implications of the standard algorithms for external sorting, selection, projection and joins.
- Describe the mechanics of both hash and B+ tree indices.
- Perform both basic query optimization and overall database tuning operations.
CSCI 340 - Networking
3 Credit Hours
Principles of abstraction underlying modern networks with an emphasis on the technologies underlying the Internet. Applications, transport protocols, routing protocols, network protocols and link protocols will be covered. Problem solving, including programming, at all layers of the networking model will be emphasized.
Prerequisites: Undergraduate level CSCI220 Minimum grade of C

Outcomes:
• Understand the terminology and basic concepts of networking deeply enough to be able to define terms and explain the workings of the network at each of the levels studied.
• Understand the role of protocols, be able to explain several common protocols, and be capable of implementing one of reasonable complexity.
• Be able to write applications at various levels of the network model.
• Demonstrate comfort discussing and exploring networking concepts.

CSCI 341 – Network Vulnerabilities, Penetration, Defense
3 Credit Hours
This course will cover the skills necessary to understand different types of attacks and exploits against hosts and networks. Students will utilize hands-on labs to learn reconnaissance, scanning, exploitation, post-exploitation, and other penetration testing methodologies used by ethical hackers. The course will also cover the computer security resources that cover new vulnerabilities and innovative methods to protect networks. Fundamentals of information assurance including legal, ethical, and compliance issues will be taught and enforced.
Prerequisites: CSCI 240, CSCI 340, MATH 156 or MATH 256

Outcomes:
• Organize and apply various open-source tools, techniques, and methods for conducting network penetration testing and vulnerability analysis
• Identify internal/external vulnerabilities and exploits and assess their security risks to computer networks, applications, and operating systems
• Utilize security design principles and emerging threats and tools to analyze systems and take action to defend the systems
• Determine the ethical implications of penetration testing and evaluate the roles of penetration testers.
• Produce a comprehensive penetration testing report that communicates legal issues, compliance regulations, analysis of a system, severity of security flaws, and prioritized recommendations that meets the needs of specific audiences.
CSCI 381 – Topics in Cybersecurity
3 Credit Hours
Advanced cybersecurity topics – Cyber Defense
In this Secure Network Design and Tactical Perimeter Defense course, students will design a secure network infrastructure and develop a security policy. Students will learn concepts, applications, and configuration of proxy servers, firewalls, Domain-based networks and HIDS/NIDS. Students will use SIEM (Security Information and Event Management) to monitor deep packet inspection/log analysis to detect network anomalies. The course also covers the basics of defending a network, and the basic tools and techniques that can be used to protect a network and communication assets from cyber threats.
Prerequisites: CSCI 240 and CSCI 340

Outcomes:
• Describe the key concepts in network defense (defense-in-depth, minimizing exposure, etc.).
• Develop a secure network infrastructure that includes various Operating Systems, Network Access Controls, DMZ, Web/Proxy servers, VPN, etc.
• Deploy and configure network defense tools (Firewalls, HIDS/NIDS, Honeypots, etc.) to defend against attacks and mitigate vulnerabilities.
• Analyze and develop security policies that can be implemented on systems to protect a network.
• Effectively monitor network traffic and learn to detect and respond to network anomalies
• Evaluate how network operational procedures are related to network security.

BAIS 359 - Hardware, Software & Security
3 Credit Hours
Provides an introductory overview of computer hardware, software and security on computers. Includes hardware components; troubleshooting, repair, and maintenance; operating systems interfaces and management tools; networking components, computer security; and operational procedures.

Outcomes:
• Describe the role and basic functions of an operating system, and how operating systems interact with hardware and software applications.
• Identify and describe basic security issues of operating systems.
• Identify various operating systems and their applications.
• Use command line interfaces to troubleshoot and repair operating and network system problems.
• Repair, optimize, configure computer hardware and software to eliminate problems.
• Use and manage file systems, operating system utilities, backup programs, and optimization.
• Describe and apply appropriate operational procedures including safety, environmental procedures, good communication skills, and professional behavior.
BAIS 459 - Fundamentals of Security

3 Credit Hours
Introduces basic computer and network security concepts and methodologies. Covers principles of security; compliance and operational security; threats and vulnerabilities; network security; application, data, and host security; access control and identity management; and cryptography.

Outcomes:
- Define the principles of cybersecurity.
- Describe why each principle is important to security and how it enables the development of security mechanisms that can implement desired security policies.
- Understand ethics of cybersecurity.
- Analyze common security failures and identify specific design principles that have been violated.
- Given a specific scenario, identify the design principles involved or needed.
- Understand the interaction between security and system usability and the importance for minimizing the effects of security mechanisms.
- Develop defense strategies and disaster recovery plans for organizations.