

Student Learning Outcomes Library

Office of Assessment & Accreditation

Indiana State University

BS Computer Science

Spring 2020

Outcome	Related Foundational Studies or Graduate Goal
1A - Basic Programming Can write and debug correct programs for problems that are straightforward (do not need advanced algorithmic techniques) but may require any of the standard programming constructs.	FS1, FS2, FS3, FS10
1B - Programming Paradigms Can program proficiently in more than one programming paradigm. This should include at least two of the following: procedural/imperative, object oriented, functional, event driven, logic based, concurrent.	FS1, FS2, FS3, FS10
2A - Data Structures Can use basic data structures (lists, stacks, queues, binary search trees, and hash tables) in writing programs and answer basic questions on efficiency of these data structures.	FS1, FS2, FS3, FS10
2B - Algorithms Analysis Can analyze the running time and correctness of standard algorithms which they have learned. Can answer basic questions and reason about graphs, discrete probability, basic set theory, and basic number theory.	FS1, FS2, FS3, FS10
3A - Architecture Can explain the basic internal workings of computer systems, including both hardware and software. Can explain tradeoffs in CPU design, and can write correct assembly code for basic tasks.	FS1, FS2, FS3, FS10
3B - Operating Systems Can explain the key problems that operating systems are meant to solve (synchronization, resource sharing, virtualization), including traditional solutions to these problems.	FS1, FS2, FS3, FS10

<p>3C - Programming Design in Specialized Areas</p> <p>Proficient at programming in specialized areas that are very common in industry, including databases, web programming, networking, and advanced system administration. Can explain the key algorithms, programming languages, and data structures used in each area, and be able to apply this knowledge to write efficient programs.</p> <p>Note: This outcome is specific to the Information Science concentration of the major.</p>	<p>FS1, FS2, FS3, FS10</p>
<p>3C - Advanced Algorithms and Theory</p> <p>Can explain various models of computation and how these are used to model and efficiently solve various computational problems. Important models include finite automata, regular expressions, context free grammars, and Turing machines. Can explain the concept of NP-completeness and its relation to intractability. Can explain typical ways to deal with hard problems, including brute-force search and some other method (e.g., backtracking search).</p> <p>Can apply the standard algorithm techniques of greedy, dynamic programming, and divide and conquer. Proficient at programming and algorithms analysis using these techniques. Proficient at programming and using basic graph algorithms such as those for shortest path and minimum spanning trees. Can apply some other algorithm techniques such as linear programming and randomized algorithms.</p> <p>Can explain the concept of program correctness proofs, and give correct proofs for standard algorithms. Can explain the concept of automated theorem proving, and be able to use automated theorem provers.</p> <p>Note: This outcome is specific to the Computing Science concentration of the major.</p>	<p>FS1, FS2, FS3, FS10</p>