



SYRACUSE UNIVERSITY

ECS 102

Introduction to Computing

2017-2018

(3 credits)

*Faculty: Ehat Ercanli, Associate Professor, Department of Electrical and Computer Engineering
Administrative Contact: Avinash Kadaji, Assistant Director of Information Systems, Project Advance*

ECS 102 is a one-semester course taught at Syracuse University to all freshmen engineering, computer science, visual and performing art students, and most students at the Newhouse School of Public Communications and the Maxwell School of Public Affairs. The course covers computing concepts, principles of programming, applications of computing concepts, and problem solving in engineering and computer science. Laboratory topics will include problem solving projects from various engineering disciplines. Acceptable languages are C++ OR Java.

Course Outline

Week 1: What is Abstraction?; Course Requirements and Course Overview; Overview of Abstraction; Modeling (Verification, Correctness)

Week 2: Computer Abstraction; Program Development (Editing, Compiling, Linking); Representing Information (Variable Declaration, Data Types); Displaying Information (Format, Output)

Week 3: Operational Abstractions; Arithmetic and Comparison Operations (+, -, *, /, % and >, <, ==, !=, >=, <=); Special Assignment Operations (assignment = / ++, - - / *=, +=, -=); Program Control Constructs (if, else, while, switch, case, break)

Week 4: Functional Abstraction; Constructs (do while, for); Functions (return value, name, parameters); Functions (pass by value, prototypes, body)

Week 5: Ordering Abstraction; Arrays (linear ordering, index); Arrays (pointer constant, passing arrays as parameters), Arrays (multidimensional)

Week 6: Indirect Abstraction; Pointers (indirect reference, addresses); Pointers (to arrays); Pointers (passing by reference)

Week 7: Hierarchical Abstraction; Strings (characters, ASCII); Strings (comparison, I/O, appending); Scope (global functions)

Week 8: Structural Abstraction; Structures (data groups, access); Structures (Structure Pointers); Structures (passing by reference)

Week 9: What is Computer Based Problem Solving?; Overview of Problem Solving; Design Paradigm – Functional Decomposition (Requirements, Operations); Design Paradigm – Functional Decomposition (Hierarchical Organization, Function Definition)

Week 10: Problem Solving – Performing Computation; Computation, Expressions (Precedence); Precision (Integer vs. Real, Type Conversion)

Week 11: Problem Solving – Displaying Information; Display Management; I/O Streams; Files (fstream)

Week 12: Problem Solving – Advanced Operations; Libraries; I/O Streams; Graphics (color, drawing, images)

Week 13: Problem Solving – the Internet; HTML; Overview of the Internet; Creating a Home Page

Week 14: Problem Solving; Dynamic Allocation (new, delete); Error Handling; Review

Week 15: Final Exam

| Title/Author (Publisher) | Price Per Copy | Ordering Source |
|---|----------------|--------------------------------|
| <i>Problem Solving with C++</i> , 9 th Edition; Savitch, W. ISBN: 9780133835267 | \$80.60 | Pearson Phone: 800-848-9500 |
| <i>Java Software Solutions: Foundations of Program Design</i> , 9 th Edition; Lewis & Loftus (Addison-Wesley) ISBN: 9780134462028 | \$101.00 | Pearson Phone: 800-848-9500 |

Prices are subject to change.