

SYLLABUS
PART I
EDISON COMMUNITY COLLEGE
CIT 222S INTRODUCTION TO PROGRAMMING USING C++
3 CREDIT HOURS

COURSE DESCRIPTION

Introduction to structured programming using the C++ programming language. Topics include control flow, data types, functions, parameter passing, input/output streams, and an introduction to object-oriented programming. Prerequisite: ENG 091D, ENG 093D, MTH 098D or equivalent, CIT 110S. Recommended Preparation: CIT 111S. Lab fee.

COURSE GOALS:

The student will:

Bloom's Level		Program Objective
2	1. Execute the software development process to create a top-down design approach when writing computer programs.	1, 2, 3, 7
2	2. Use C++ functions, procedures, and other features listed in the topics.	1, 2, 3, 7
2,3	3. Use a team approach to solve a computer programming problem.	1, 2, 3, 6
5	4. Create appropriate documentation within C++ programs.	1, 2, 9
2, 5	5. Use the features of object-oriented programming to design C++ programs using classes and objects.	3,4,7,12
4	6. Analyze program compiler statements, and implement appropriate corrective actions for error-trapping.	1, 2, 3, 7,8
2	7. Discuss the importance of ethics in the computer industry, and the role they play in the field of computer games design and programming.	1, 2, 10, 13

CORE VALUES

The Core Values are a set of principles which guide in creating educational programs and environments at Edison. They include communication, ethics, critical thinking, human diversity, inquiry/respect for learning, and interpersonal skills/teamwork. The goals, objectives, and activities in this course will introduce/reinforce these Core Values whenever appropriate.

TOPIC OUTLINE

1. Creating a C++ Program
2. C++ Identifiers/Default
3. C++ Keywords
4. Basic Data Types
5. Literals, Variables, and Constants
6. Storage Modifiers
7. Enumerated types
8. Character Strings
9. Printing Messages to the Screen
10. Reading Data from the Keyboard
11. Assignment Operators
12. Mathematical Operators (including static_cast Operator)
13. Boolean Expressions and Logic Operators

14. Relational Operators
15. Shorthand Assignments
16. Operator Priority
17. Block Scope and Local Variables
18. Variables within Embedded Scopes
19. Global Variables
20. Boolean Expressions (if Statement, Conditional Operator, switch Statement)
21. Loops (while Loop, do-while Loop, for Loop)
22. Controlling Loops with continue and break
23. C++ Library Structure
24. Calling Library Functions
25. Function Structure
26. Function Prototypes
27. Modularization of Programs
28. Default Parameters
29. Passing Parameters by Reference
30. constParameters
31. Overloading Functions
32. Header Files
33. Handling Multiple Source Files
34. extern Keyword and Global Variables
35. Inline Functions
36. Single-Dimension Arrays, Searches, and Sorts
37. Strings
38. Finding Tokens in Strings
39. Multidimensional Arrays
40. Array Parameters (Single and Multidimensional)
41. Recursion
42. Structures and Unions
43. Pointers
44. Void Pointers
45. Pointer Arithmetic
46. Pointers in Functions
47. Pointers and Arrays
48. Pointers versus References
49. Classes
50. Encapsulation
51. Abstraction
52. Inheritance (Is-A and Has-A)
53. Polymorphism
54. Composition
55. Class Declaration and Implementation
56. Object Scope
57. Using friend Classes
58. Constructors
59. Destructors
60. Function Overloading
61. Operator Overloading
62. Copy Constructor and This Pointer
63. Dynamic Objects (new and delete Operator)
64. Memory Leaks

65. Multiple Inheritance
66. Function Binding
67. Abstract Base Classes
68. Uniform Modeling Language
69. Object Hierarchy
70. Text Streams
71. Random Access Files
72. Object Serialization
73. Setting Up Simple DirectMedia Layer (SDL)
74. Using SDL for Input and Sound
75. Single Linked-Lists
76. Templates
77. Assert and Exception Handling
78. #define and #undef
79. Standard Template Library (STL)
80. Containers and Iterators