San Bernardino Valley College
Curriculum Approved: January 24, 2005

I. CATALOG DESCRIPTION:

A. Department Information:

Division: Science and Math
Department: Computer Science
Course ID: CS 190
Course Title: Programming in C++
Units: 4
Lecture Hours: 3
Laboratory Hours: 3
Prerequisite: CS 110

B. Catalog and Schedule Description:
An introduction to the object-oriented language, C++. Topics include object-oriented design, program logic structures, problem solving techniques, arrays and records, procedures and functions, classes and objects, pointers and references, inheritance, and polymorphism.

II. NUMBER OF TIMES COURSE MAY BE TAKEN FOR CREDIT: One

III. EXPECTED OUTCOMES FOR STUDENTS:

Upon completion of the course the student should be able to:

A. Design an algorithm for a programming problem;
B. Define the vocabulary for object-oriented programming;
C. Use object-oriented methods;
D. Read and write C++ programs;
E. Document a program;
F. Use procedures and functions in modular programming;
G. Distinguish and use selective logic;
H. Compare the performance of different types of loops;
I. Organize data using arrays and records;
J. Use pointers for linked lists;
K. Apply design standards in software engineering;
L. Transfer the techniques learned in this course into another programming course.

IV. COURSE CONTENT:

A. Overview of programming and problem solving
   a) Computers and programming languages
   b) HIPO analysis of a problem
   c) Top-down approach and stepwise refinement
   d) Divide and conquer
   e) Algorithmic problem solving
   f) Object-oriented programming
B. Elements in C++
   a) Syntax and semantics
   b) Data and data types
   c) Statements and comments
   d) Constructing blocks
C. Program development
   a) Declarations
   b) Input streams
   c) Arithmetic expressions
   d) Function calls and library functions
   e) Output formats
D. I/O designs

a) Interactive I/O
b) Noninteractive I/O
c) File I/O
d) Error handling

E. Selective logic
a) Logical expressions and operations
b) If/Then and If/Then/Else statements
c) Nested statements

F. Loops
a) While loops
b) For loops
c) Event-controlled loops
d) Nested loops

G. Functions
a) Modular design
b) Declarations and definitions
c) Parameters
d) Scope and lifetime of variables
e) Interface design

H. Built-in and user-defined data types
a) Numeric character
b) Character data
c) Enumeration types
d) Header files
e) Type correction

I. Arrays
a) Declaring and accessing rays
b) Passing arrays as parameters
c) Parallel arrays
d) Multidimensional rays

J. Lists and strings
a) Ordered lists and unordered lists
b) Sorting and searching lists
c) String initialization
d) String I/O

K. Records
a) Record design
b) Hierarchical records
c) Arrays of records

L. Object-oriented development
a) Classes
b) Objects
c) Specification and implementation files
d) Inheritance
e) Composition
f) Dynamic binding and virtual functions

M. Software engineering
a) Algorithm efficiency
b) Maintenance and evolution

V. METHODS OF INSTRUCTION:

A. Lecture
B. Discussion
C. Multi-media
D. Projects

VI. TYPICAL ASSIGNMENTS:

A. Read the chapter reviewing the basics of computer programming and outline the main points. Email your outline to the instructor before the next laboratory meeting.
B. Analyze programming problems and design algorithms
C. Write C++ programs in lab
   1. Sample programming problem using C++ class:
      a) Write a class *parser* that can:
         i) Read in string input from the user
         ii) Parse the line one word at a time. That is, isolate and store the next word in the line
         iii) Return a pointer to the currently parsed word
         iv) Return where in the line the first character of the word id located relative to the start of the line
      b) A word is defined to consist of letters, numbers, and underscores. Letters are case sensitive
      c) To test the class, write a *main* ( ) function that merely reads in strings and outputs the words one at a time showing where the first character is in the line
D. Discuss special C++ programming techniques in class

VII. EVALUATION(S):

A. Programming projects: One project per week
B. Examinations and quizzes
   1. Two exams: midterm and final
   2. Weekly quizzes on reading assignments
   3. Sample test questions:
      a) In what ways does a class differ from a structure? In what ways are they similar?
      b) What are the two types of member functions?
      c) How does a class declaration differ from a class definition?
      d) When would you define a member function *outside* the class definition?
      e) When would you define a member function *inside* the class definition?

VIII. TYPICAL TEXT(S):

1. C++ for Engineers and Scientists, Bronson, Incorporated Course Technology, 2005.
2. C++ Programming: Programming Design with Introduction to Data Structures, Malik, D.S., Incorporated Course Technology, 2004.
3. C++ How to Program, 4th edition, Deitel, Harvey and Deitel, Paul; Prentice Hall, 2002.

IX. OTHER SUPPLIES REQUIRED OF STUDENTS: None