

Fall 9-1-2005

CRT 270.02: C++ Programming

Rhonda Tabish

University of Montana - Missoula College, rhonda.tabish@umontana.edu

Let us know how access to this document benefits you.

Follow this and additional works at: <https://scholarworks.umt.edu/syllabi>

Recommended Citation

Tabish, Rhonda, "CRT 270.02: C++ Programming" (2005). *Syllabi*. 9956.
<https://scholarworks.umt.edu/syllabi/9956>

This Syllabus is brought to you for free and open access by the Course Syllabi at ScholarWorks at University of Montana. It has been accepted for inclusion in Syllabi by an authorized administrator of ScholarWorks at University of Montana. For more information, please contact scholarworks@mso.umt.edu.



The University of Montana

College of Technology

Fall Semester 2005

CRT270 C++ Programming

Prerequisite: CRT121

Credits 3

Rhonda Tabish

Rhonda.Tabish@umontana.edu

243-7808

Office hours by appointment

Sec 1 – M,W 10:10 – 11:00, F 11:10 – 1:00 HB04

Sec 2 – M,W 9:10 – 10:00, F 9:10 – 11:00 HB04

COURSE DESCRIPTION:

Object oriented programming using C++. Implementation of structured programming concepts along with construction classes to create data types for defining objects.

STUDENT PERFORMANCE OBJECTIVES:

1. Students will create, compile, and debug structured source code.
2. Students will design and implement the user interface in a command line environment using an object oriented programming language.
3. Students will design and implement logical expressions containing operators, variables, and constants.
4. Students will design and implement the standard programming constructs of loops, repetition, and conditionals to solve problems.
5. Students will structure tasks into logical functions and procedures.
6. Students will create and manage variable and array structures for storing information.
7. Students will design and implement multiple methodologies for the user to input and output information with a computing system.
8. Students will demonstrate effective documentation techniques.
9. Students will construct efficient error control structures.

TEXT:

A First Book of C++, Second Edition, by Gary J. Bronson

EVALUATION:

A final grade will be determined by total points received on assignments and quizzes in relationship to total points available.

Assignments **will not** be accepted after **4:00** p.m. of the date due. In class exercises cannot be made up.

Makeup tests **are not** offered. Emergency situations are handled privately on a case by case basis.

GRADING SCALE:

93 - 100	A
87 - 92	B
80 - 86	C
74 - 79	D

FINAL EXAM:

Sec 1 – Friday, December 16 8:00 – 10:00

Sec 2 – Thursday, December 15 8:00 – 10:00

ACADEMIC INTEGRITY:

Students are expected to follow the University of Montana Student Code. The code includes the following:

Academic Misconduct:

... Academic misconduct is defined as all forms of academic dishonesty, including but not limited to:

Plagiarism: Representing another person's words, ideas, data or material as one's own.

Substituting or arranging substitution, for another student during an examination or other academic exercise: Knowingly allowing others to offer one's work as their own.

Student Code copies are available at Student Services or www.umt.edu/studentaffairs/

DISABILITY ACCOMMODATION: Eligible students with disabilities will receive appropriate accommodations in this course when requested in a timely way. Please speak with me after class or in my office. Please be prepared to provide a letter from your DSS Coordinator.

CRT270 C++ Programming

COURSE OUTLINE

- I. Introducing C++
 - A. UNIX Review
 - B. Create, Compile and Debug a C++ Program
 - C. Function and Class Names
 - D. The main() Function
 - E. Programming Style

- II. Logical Expressions with Operators, Variables, and Constants
 - A. Data Types
 - B. Arithmetic Operators
 - D. Variables and Declarations
 - E. Integer Qualifiers
 - F. Const Qualifier

- III. Creating a User Interface
 - A. User Input
 - B. The cin Object
 - C. Program Output
 - D. The cout Object
 - E. String Fundamentals

- IV. Conditionals, Error Control
 - A. Relational Expressions
 - B. The if/else Statement
 - C. Nested if Statements
 - D. The switch Statement
 - E. Error Control

- V. Loops and Repetition
 - A. The while Statement
 - B. cin Within a while Loop
 - C. Using Sentinels
 - D. The for Statement
 - E. cin Within a for Loop
 - F. The do Statement

- VI. Logical Functions and Procedures
 - A. Function and Parameter Declarations
 - B. Returning a Single Value
 - C. Pass by Reference
 - D. Variable Scope and Storage Class

VII. Array Structures

- A. One-Dimensional Arrays
- B. Array Initialization
- C. Arrays as Arguments
- D. Two-Dimensional Arrays

VIII. Pointers

- A. Addresses and Pointers
- B. Array Names as Pointers
- C. Pointer Arithmetic
- D. Passing Addresses
- E. Pointers and Library Functions
- F. Pointer Arrays

X. Records as Data Structures

- A. Single Records
- B. Arrays of Records
- C. Record Structures as Function Arguments
- D. Linked Lists
- E. Dynamic Structure Allocation
- E. Unions