

Course Outcomes	CSC 1253
------------------------	-----------------

CSC 1253: Computer Science I with C++

Credit Hours: 3 hours

Frequency: Fall and Spring semesters

Prerequisites:

Credit or registration in MATH 1550 or credit in MATH 1431. Credit will not be given for both this course and CSC 1248 or 1250 or 1350 or ISDS 3107.

Prerequisites by Topics:

Basic math skills and problem solving ability

Catalog Course Description:

Fundamentals of algorithm development, program design and structured programming using an object-oriented language.

Course Outcomes

1. Be exposed to basic hardware and software concepts
2. Be familiar with issues related to software design
3. Master using key structured programming constructs: declarations, sequence, selection, repetition, evaluating expressions.
4. Be familiar with using C++ functions and the concepts related to good modular design.
5. Master one-dimensional and two-dimensional arrays
6. Be familiar with using C++ structures.
7. Be familiar with using pointers and reference parameters.
8. Be familiar with using text file input/output
9. Be familiar with C++ classes.

Texts and Other Course Materials

- Foundations in C++ - Walter Savitch 0-536-94800-3 PB Latest Addison

Major Topics

- Hardware and Software: different types of computers and terms; parts of a computer (memory, CPU, I/O); how data and instructions are represented including base conversions; types of software; types of programming languages and terms. Introduction to operating systems.
- Compilation process, programming process (software development

lifecycle), algorithm development, structured programming principles and construction. Writing algorithms using pseudocode.

- Preprocessor directives and system libraries.
- Variables, expressions (arithmetic, Boolean, literals), assignment statements, precedence, association, data types, enumeration types, union types.
- Interactive input/output, formatting output.
- Branching statements (single selection, multi-way, switch).
- Repetition structures (while, do-while, for). Counter-control loops (incremental and decremental), event control loops including sentinel control. Nested repetition.
- Programming standards and style guidelines: good documentation.
- Pointers including pointers to pointers and references.
- Modular design, functions, predefined functions. (omit recursive functions). Call-by-value, call-by-address, call-by-reference. Overloading a function name.
- Single dimensional and multi-dimensional arrays. Partially filled arrays.
- Unordered and ordered linear search and binary search of arrays.
- Sorting of arrays: bubble, insertion and selection sorts.
- Structures and arrays of structures. Hierarchical structures.
- Character and string I/O. Predefined string functions. Character manipulation I/O: get and put. Introduction of the standard class string.
- Streams and text File I/O: open files, close files, end of file, and formatting output.
- Preview of classes (encapsulation): Public and private members; accessor and mutator functions; static members.
- Preview of Inheritance and templates.
- Preview constructors and destructors.
- Preview overloading.
- Preview arrays of classes.

Assignments/Projects/Laboratory Projects/Homework

- Using the C++ compiler and the UNIX system
- Writing a simple C++ program using data declarations, conditionals and loops
- Designing and implementing a modular solution
- Designing and implementing a solution using 1-d and 2-d arrays

Curriculum Category Content (estimated in semester hours)

Area	Core	Advanced	Area	Core	Advanced
Algorithms	10%		Data Structures	30%	
Software Design	20%		Prog. Languages (Syntax)	40%	
Computer Arch.	3 hrs				

Relationship to Criterion 3 Outcomes

A	B	C	D	E	F	G	H	I	J	K
*	*	*		*	*	*		*		*

Math Fundamentals:

Data Structures:

1-d arrays (6 hrs), 2-d arrays (6 hrs), arrays of structures or classes (6 hrs)

Algorithms and Software:

Analysis and Design (14 hrs)

Computer Organization and Architecture:

Computer classifications, peripherals, networking concepts

Computer hardware including the CPU (total 3 hrs)

Concepts of Programming Languages:

Concepts of imperative languages and procedural programming,

Syntax and semantics of C++ (total 20 hrs)

Social and Ethical Issues:

Computer human interaction and the importance of critical system verification (3 hrs)

Oral Communication (presentations) – none

Written Communication:

Documentation of programs (6 hrs)

Course Coordinator: Barbara Guillott

Last Modified: April 26, 2007