

UNIVERSITY *of* MARYLAND
UNIVERSITY COLLEGE - *Asian Division*

CMIS 102 Introduction to Algorithm Design and Programming

Credits: 3

Monday through Thursday 16:40 to 17:55, Iwakuni MCAS, Japan

Prerequisites: None but there will be some math in this course.

Instructor: Robert Laurie

Email: rlaurie@ad.umuc.edu

Web Site: <http://www.islandman.org>

Textbooks:

1. *Programming & Problem Solving with C++*, 2nd Ed.
Dale/Weems/Headington, ISBN#: 0763710636
2. *Simple Program Design*, Robertson, ISBN#: 61901590X, Optional

Description:

A study of techniques for finding solutions to problems using structured programming and step-wise refinement. Topics include principles of programming, the logic of constructing a computer program, and the practical aspects of integrating program modules into a cohesive whole. Algorithms are used to demonstrate programming as an approach to problem solving, and basic features of the C ++ language are illustrated. (May not be applied to a major in computer and information science.)

Objectives:

1. Define and practice heuristic learning.
2. List methods to improve your critical thinking, as well as how to improve your learning and memory skills.
3. List ways to improve comprehension in what you read and hear.
4. Demonstrate improved skills in solving math and logic problems.
5. Develop confidence in your own ability to solve problems.
6. Define and understand the following programming terms: source code, object code, and compiler.
7. Distinguish between a syntax error and a runtime error.
8. Conceptually define the three types of program statements: sequential, conditional and iterative.
9. Understand the process of stepwise refinement in developing programs to solve problems.
10. Demonstrate understanding of programming statements written in structured pseudo-code.
11. Write basic programs in C++; compile them; and execute them.

Exams:

Exams 1 and 2 will cover topics discussed in that portion of the class. The final exam will be comprehensive and cover topics discussed throughout the course. I encourage students to study together for exams and will not curve scores.

The examination days are listed on the course schedule by week number. Only students with officially excused absences will be able to make up the exams, others will receive a grade of zero. You must contact me via email, for me to authorize a makeup exam time prior to the scheduled exam time. You need to provide me with documentation verifying the excused absence. Failure to comply with these requirements will result in a score of zero on the exam.

There will be no extra credit awarded in the course, so do your best on the given assignments.

Grades:

The grade in the course will be based on 3 exams and several projects:

Items	Scores	Percent
Exam 1	100	20%
Exam 2	100	20%
Final Exam	150	30%
Projects	150	30%
Total	500	100%

Grade	Scores	Percent
A	500 to 450	100.0 to 90.0%
B	449 to 400	89.9 to 80.0%
C	399 to 350	79.9 to 70.0%
D	349 to 300	69.9 to 60.0%
F	< 300	Less than 60%

Projects:

Project assignments of various point values will be given throughout the term. Completed project reports must be submitted on the due dates. Late assignments will be reduced 25% of the total point value for each class period late. No projects will be accepted after the final exam time.

Grading will be 80% objective (results, explanations, conclusions) and 20% subjective (neatness, clarity, conciseness, extra work). A project report that minimally meets all specifications will receive a score of 80% of the total points. If any portion of a project is plagiarized, the entire project will receive a score of zero.

Attendance:

Class attendance is mandatory. If you miss a class meeting, it remains your responsibility to obtain information concerning the material covered and upcoming assignments. Excessive absences may result in your being assigned a grade of *F(n)!* *Failure due to non-attendance.*

CMIS 102 Course Schedule (Tentative *)

Date:	Topics:	Read Before Class:
Week 1	Reasoning and problem solving, Problem Solving and Programming	Dale: Preface, Ch 1 Robertson: Ch1
Week 2	C++ Syntax and the Program Development Process.	Dale: Ch 2 Robertson: Ch 2, 3
Week 3	C++ Expressions, Data types, and Output EXAM 1 (Thursday Class)	Dale: Ch 3
Week 4	Input and Software Design Process Conditions and Selection Structures	Dale: Chap 4 & 5
Week 5	Logic Operators	Dale: Chapter 5
Week 6	Loop Structure, EXAM 2 (Thursday Class)	Dale: Ch 6 Robertson Ch 5,6
Week 7	More Loops, File I/O, and OO Design	Dale: Ch 6, 4, 2
Week 8	FINAL PROJECT DUE (Tuesday class) FINAL EXAM (Last Class)	

* This syllabus is tentative and is subject to change.