CMSC 11600-1
Introduction to Programming II

Instructor: Catalin Dumitrescu
Office: Ryerson 403#c16
Email: cldumitr@cs.uchicago.edu
Office Hours: MW 15:30-16:30

TAs: Irina Matveeva, Zhimin Xie
Offices: Ryerson 178, 166
Emails: matveeva@cs.uchicago.edu, zxie@cs.uchicago.edu
Office Hours: Wednesdays 4:00pm - 6:00pm, Thursdays 5:00pm - 7:00pm

Course Overview: CMSC11600 is an introduction to computer programming in the C++. You will use g++ under GNU/Linux or CodeWarrior under MacOS or Windows to compile your programs. You can meet TAs or me if you want additional help getting started. I will expect you have read something before you come in and/or attended CodeWarrior and g++ tutorials that will be held at the beginning of the second week of classes in the CS Labs, located on Level A in Regenstein library.

Prerequisite: CMSC105 or CMSC115

Books:
- “C++ Primer”, 3rd edition, by S. Lippman and J. Lajoie, ISBN# 0201824701 (OPTIONAL)

Class Webpage: http://www.classes.cs.uchicago.edu/classes/archive/2002/winter/cs11600-1/

Discussion List: cs11600-1@cs.uchicago.edu
List-Help: mailto:cs11600-1-request@cs.uchicago.edu?subject=help
List-Subscribe: http://mailman.cs.uchicago.edu/mailman/listinfo/cs11600-1
or mailto:cs11600-1-request@cs.uchicago.edu?subject=subscribe
List-Archive: http://mailman.cs.uchicago.edu/mailman/cs11600-1/

Tentative Schedule:
- Basics:
  2. (2.1 Jan 7) Introduction to C/C++. Program structure. Basic Statements. Basic notions about Source Files, Preprocessor, Compilation and Linking Processes

- non-OO Programming:
  3. (2.2 Jan 9) Types. Expressions. Functions. Arrays
  4. (2.3 Jan 11) Pointers. Pointers to arrays and functions. Dynamic memory allocation, misc. about non-OO syntax
5. (3.1 Jan 14) Structures and Unions.
I/O operations using printf or streams (?)

• OO Programming:
6. (3.2 Jan 16) ADT Introduction. Encapsulation
7. (3.3 Jan 18) Encapsulation (2). Class Members
8. (4.1 Jan 21) Static, Virtual and Friend Members
9. (4.2 Jan 23) Constructors / Destructors. Copy Constructors
10. (4.3 Jan 25) Function and Class Member Overloading
11. (5.1 Jan 28) Midterm I

• Generic Programming:
12. (5.2 Jan 30) Function and Class Templates
13. (5.3 Feb 1) Standard Template Library

• OO Programming (2):
16. (6.3 Feb 8) Polymorphism
17. (7.1 Feb 11) Undergraduate Break
18. (7.2 Feb 13) Polymorphism (2). Exceptions
19. (7.3 Feb 15) Exceptions. Exception Handling (2)
20. (8.1 Feb 18) Examples: List, Queue, Stack, Tree ADTs. More about STL (1)
21. (8.2 Feb 20) Debugging and Testing
22. (8.3 Feb 22) Example: Simple communication library in C++ implemented over TCP/IP
23. (9.1 Feb 25) Midterm II
24. (9.2 Feb 27) Iterators. Notion of algorithm costs (1). More about STL (2)
25. (9.3 March 1) Notion of algorithm costs, big-O notation, sort algorithms (2)

• Applications:
26. (10.1 March 4) A simple Scheme interpreter implemented in C++
27. (10.2 March 6) Final review

Homework: weekly problems
Each set of problems will be posted one week before the due date on Wednesday, excepting the first homework (Friday). The due dates are Thursdays at 8:00 pm. All problems must be submitted using hwsubmit script installed ONLY on Linux machines (CS or Regenstein). All problems and programs must be handed in plain text format.

• Required format for homework:
1. Each problem must be solved in a separate directory, named ProblemNumber (e.g. Problem1, Problem2).
2. Each file must be named using the following convention:
   firstName_hwNumber_problemNumber_shortdescription.{cc|hh}
   Example:
   – catalin_1_1_main.cc, catalin_1_1_queue.cc, catalin_1_1_queue.hh
   – catalin_1_2_main.cc, catalin_1_2_stack.cc, catalin_1_2_stack.hh
All files containing a main function must have the description “main”.
3. Each file must contain at the very beginning:
   // Your Full Name, Your Student ID, Your Preferred Email Address
   // Homework number, Problem number (e.g. Homework 1 / Problem 1)
   // Your Honesty Pledge (Example: I pledge that this homework has been done in accordance with the University’s Academic Honesty policy.)
   // A List of Known Bugs in Your Solution
• Grading Policies:
  Problems which cannot be tested (compilation errors) receive at most half of the credit.

• Regrading Policy:
  If you believe an error was made in the grading of your program, please submit a request to cl Dumitr@cs.uchicago.edu
  no more than a week after the graded homework is made available to you!

**Exams:** Midterm I is on January 25, midterm II is on February 18 and the final is on March ??.
Exam questions will be based on discussions and examples covered in class or assigned as homework problems.
Attendance at midterms and final exam is required!

**Final Grade:** 50% Weekly Homework, 10% Midterm 1, 15% Midterm 2, 15% Final, 10% Class Participation (a
somehow subjective part of your grade)

**Academic Honesty:** Programs, homework and examinations are to be completed without collaboration with any
other person. However, you may discuss with your colleagues the general nature of the solutions to programs and
homework problems. Violations of this policy will be reported and, in addition, the student will receive a zero grade
on the program, homework or examination.

**Others:** (related to class)
1. Regular attendance at lectures and tutorials is strongly recommended!
2. No late assignments will be accepted without a strong motivation!
**PLEASE DO NOT ASK FOR EXTENDED SUBMISSION!**
3. Check the class website and discussion list before each lecture.