

Introduction to C++ Programming

ITP 165 (2 Units)



Fall 2014

Objective

This course will teach students problem solving skills through the use of the C++ programming language.

Concepts

Programming fundamentals including variables, control statements, loops, and arrays, pointers, functions and object-oriented programming.

Prerequisites

None. This class is intended for students with no prior programming experience.

Instructor

Listed on Blackboard under Contacts

Contacting the Instructor

Listed on Blackboard under Contacts

Office Hours

Listed on Blackboard under Contacts

Lab Assistants

Listed on Blackboard under Contacts

Lecture / Lab

See online schedule of classes

Required Textbooks

None.

Optional References

To supplement course material a good introductory C++ reference is recommended like the tutorials available at LearnCpp.com, CPlusPlus.com, and CProgramming.com.

Course website

All course material will be on Blackboard (<http://blackboard.usc.edu>).

Grading

The following percentage breakdown will be used in determining the grade for the course.

Lab practicals	20%
Assignments	40%
Midterm exam	20%
Final exam	20%
Total	100%

Grading Scale

The following shows the grading scale to be used to determine the letter grade.

93% and above	A
90% - 92%	A-
87% - 89%	B+
83% - 86%	B
80% - 82%	B-
77% - 79%	C+
73% - 76%	C
70% - 72%	C-
67% - 69%	D+
64% - 66%	D
63% and below	F

Policies

Exams

No make-up exams (except for documented medical or family emergencies) will be offered nor will there be any changes made to the Final Exam schedule. There are two exams: a midterm and a final, and these exams are comprehensive of all topics covered.

Assignments

Each assignment must be completely *individually*. There are not any group projects in this class. The assignments will be posted on Blackboard under the “Assignments” section. Each lab assignment will include instructions, a due date, and a link for electronic submission. Labs must be submitted using this link.

It is your responsibility to submit your assignments on or before the due date. Assignments turned in one day late will have 20% of the total points deducted from the graded score. Assignments turned in two days late will have 50% of the total points deducted from the graded score. After two days, submissions will not be accepted and you will receive a 0. All assignments will be digitally submitted through Blackboard except where specifically specified. Do not email them to the lecturer or lab assistant.

If you have questions about any of the lab assignments, attend a lab session or send an email to the lab assistant assigned to the lab session in which you are registered. Do not send any email to the instructor regarding labs or ask specific lab questions during the lecture sessions. You are encouraged to attend the instructor’s office hours for lab-related questions.

Lab facilities

You are required to save your labs using a USB flash drive or a website such as [Dropbox](#). You must keep a copy of all labs. You will not be able to save your work on the ITP lab computers. Any work saved to the computer will be erased after restarting the computer. ITP is not responsible for any work lost.

ITP will have open lab hours starting the second week of the semester. The open labs will not have a lab assistant for this specific class. These lab times are there in case you need extra time to complete a lab.

Lab practicals

There will be lab practicals after most lectures. These practicals will be immediate application of the material presented in lecture. These practicals will be graded as pass/fail. For credit on each practical you must complete the practical before class time has ended and demonstrate your working code to the course staff. Each practical will contribute to your overall grade. There is no way to make up a missed practical, however a practical grade can be dropped provided either prior instructor approval or a documented emergency.

Incomplete and Missing Grades

Excerpts for this section have been taken from the University Grading Handbook, located at <http://www.usc.edu/dept/ARR/grades/gradinghandbook/index.html>. Please see the link for more details on this and any other grading concerns.

A grade of Missing Grade (MG) “should only be assigned in unique or unusual situations... for those cases in which a student does not complete work for the course before the semester ends. All missing grades must be resolved by the instructor through the Correction of Grade Process. One calendar year is allowed to resolve a MG. If an MG is not resolved [within] one year the grade is changed to [Unofficial Withdrawal] UW and will be calculated into the grade point average a zero grade points.

A grade of Incomplete (IN) “is assigned when work is no completed because of documented illness or other ‘emergency’ **occurring after the twelfth week** of the semester (or 12th week equivalency for any course scheduled for less than 15 weeks).”

Academic Integrity

USC seeks to maintain an optimal learning environment. General principles of academic honesty include the concept of respect for the intellectual property of others, the expectation that individual work will be submitted unless otherwise allowed by an instructor, and the obligations both to protect one’s own academic work from misuse by others as well as to avoid using another’s work as one’s own. All students are expected to understand and abide by these principles. *Scampus*, the Student Guidebook, contains the Student Conduct Code in Section 11.00, while the recommended sanctions are located in Appendix A: <http://www.usc.edu/dept/publications/SCAMPUS/gov/>. Students will be referred to the Office of Student Judicial Affairs and Community Standards for further review, should there be any suspicion of academic dishonesty. The Review process can be found at: <http://www.usc.edu/student-affairs/SJACS/>.

In this class, all code submissions will be ran against current, previous, and future students using MOSS, which is a code plagiarism identification tool. If your code significantly matches another student’s submission, you will be reported to SJACS.

Generally it is acceptable to discuss solutions to problems with other students, but looking at another’s code often becomes cheating. It does not matter if this code is online or from a fellow student it is cheating in all situations. Do not share your code with anyone else in this or a future section of the course. Providing your code to another carries the same penalty as copying the code yourself.

Students with Disabilities

Any student requesting academic accommodations based on a disability is required to register with Disability Services and Programs (DSP) each semester. A letter of verification for approved accommodations can be obtained from DSP. Please be sure the letter is delivered to your course instructor (or TA) as early in the semester as possible. DSP is located in STU 301 and is open from 8:30am to 5:00pm, Monday through Friday. Website and contact information for DSP

http://sait.usc.edu/academicsupport/centerprograms/dsp/home_index.html (213) 740-0776 (Phone), (213) 740-6948 (TDD only), (213) 740-8216 (FAX) ability@usc.edu

Emergency Preparedness/Course Continuity in a Crisis

In case of emergency, when travel to campus is difficult, if not impossible, USC executive leadership will announce a digital way for instructors to teach students in their residence halls or homes using a combination of the Blackboard LMS (Learning Management System), teleconferencing, and other technologies. Instructors should be prepared to assign students a “Plan B” project that can be completed ‘at a distance.’ For additional information about maintaining your classes in an emergency, please access:

<http://cst.usc.edu/services/emergencyprep.html>

Introduction to C++ Programming

ITP 165 (2 Units)

Course Outline

Note: Schedule subject to change

Week 1

Introduction

- What is a computer
- What is C++
- Numeric variables
- Basic arithmetic

Assignment

Homework 0: IDE setup (due Week 1)

Input and output

- Input and output
- Character variables
- Strings

Assignment

Homework 1: Mad Libs (due Week 2)

Week 2

C++ logic

- Boolean variables
- Comparisons
- Logical operators
- Lab practical #1

Decisions

- If, else, and else if statements
- Switch statements
- Homework 1 due
- Lab practical #2

Assignment

Homework 2: Temperature converter (due Week 3)

Week 3

Introduction to binary

- Binary numbers
- Binary operators
- Lab practical #3

Binary continued

- Binary operators (continued)
- Hexadecimal numbers
- Homework 2 due
- Lab practical #4

Assignment

Homework 3: TBD (due Week 4)

Week 4

Repetition

- Loops
- Lab practical #5

Repetition, again

- Loops (continued)
- Lab practical #6
- Homework 3 due

Assignment

Homework 4: Calculator (due week 5)

Week 5

Arrays

- Arrays
- Array traversal
- Lab practical #7

C-style strings

- Arrays of characters
- Lab practical #8
- Homework 4 due

Assignment

Homework 5: Letter counter (due week 6)

Week 6

Using functions

- Introduction to functions
- Using functions
- Lab practical #9

Creating functions

- Writing functions
- Midterm review
- Homework 5 due

Week 7

Midterm

- 15% of total grade

Examining function input

- Pass by value
- Pass by reference
- Lab practical #10

Assignment

Homework 6: Roulette (due week 8)

Week 8

Classes

- Introduction to classes
- Lab practical #11

Classes (continued)

- The std::string class
- Lab practical #12
- Homework 6 due

Assignment

Homework 7: Input sanitizer (due week 9)

Week 9

I/O streams

- Interactive I/O
- Introduction to file I/O
- Lab practical #13

File I/O

- File I/O continued
- Lab practical #14
- Homework 7 due

Assignment

Homework 8: Cipher (due week 11)

Week 10

Memory management

- Variables in memory
- Reference variables
- Lab practical #15

Pointers

- Pointer variables
- Lab practical #16

Week 11

Pointers (continued)

- Arrays and pointers
- The heap and the stack
- Lab practical #17

Dynamic memory

- Keywords new and delete
- Lab practical #18
- Homework 8 due

Assignment

Homework 9: TBD (due week 13)

Week 12

Structures

- Using structs
- Writing structs
- Lab practical #19

Classes revisited

- Writing classes
- Lab practical #20

Week 13

Pointers revisited

- Pointers and structs
- Pointers and classes
- Lab practical #21

Object oriented programming

- Introduction to object oriented programming
- Lab practical #22
- Homework 9 due

Assignment

- Homework 10: TBD (due week 15)

Week 14

Class hierarchies

- Inheritance
- Lab practical #23

Functions revisited

- Overloading functions
- Virtual functions
- Lab practical #24

Week 15

Debugging

- Debugging tools
- Debugging approaches
- Lab practical #25

Wrap up

- Advanced topics in C++
- Final review
- Homework 10 due

Final Exam

According to the final exam schedule on the Schedule of Classes