1. Course objectives
The objectives of this course are to introduce computer science fundamentals using C++, provide the ability to read and write C++ programs, designing algorithms to solve problems and implementing them using C++.

2. Course description
C++ is one of the most widely used programming languages for solving problems. The course provides an introduction to C++ from basic built-in data types and functions to user defined functions and data types. Class construct integrates the data types with functions to define new data types. The course first introduces the programming environment and then the basic techniques to solve problems. This will be followed by the topics such as algorithm development, sequencing, selection, iteration, functions, arrays, structures, classes and several other advanced topics to provide the basic concepts, working vocabulary and appropriate knowledge necessary for further studies.

Learning by doing is the method used in this course. Active participation of students in the classroom is expected in each lecture hour. The learning process consists of lectures, reading programs, writing programs, finding errors, testing, programming laboratories, reading assignments, problem solving sessions, written tests and programming tests. Project laboratory will play a major role in this course to understand the material. Numerous examples will be used to demonstrate problems, algorithm design and how to develop programs. You are encouraged to learn the concepts and system tools from all the sources available. Make sure to write your own answers to the questions and programming code.
This course covers guidelines of the ACM's Curriculum Committee.

The course is intended for Computer science, mathematics and business majors. It will be of great assistance to management and other majors who require access to practical material on the application programming.

Participants will learn the fundamentals and strategies of general problem solving, algorithm design and C++ programming.

3. Textbooks
Engaged Learning for Programming in C++ by Roberge et al., JB publishing
Programming and Problem Solving with C++ , Second edition 1999 by Dale, Weems and Headington, Jones and Bartlett publishing
C++ How to Program by Deitel & Deitel, Prentice Hall Publishing

4. Lecture + Laboratory Outline and Schedule

Days 1&2
Introduction, cout, data types, variables, cin
  Unix environment,
  Editing, executing and testing C++ programs with cout, data types, variables, cin
  Reading assignment - Chapter 1

Arithmetic
  interactive Input and Output,
  Arithmetic expressions,
  Reading assignment - Chapter 2
  Homework assignment
  Programming Lab assignment
  Problem solving session

Selection - logical operators, expressions, if else switch
  If else control structures,
  logical operators and expression,
  compound statements,
  Switch statements,
  Boolean expressions
Quiz

Days 3&4

Repetition - iteration with for while

Counter controlled loops,
event controlled loops,
Combined counter/event controlled loops,
nested loops
Reading assignment - Chapter 6, 9
Homework assignment
Programming Lab assignment
Problem solving session

Functions for problem solving

Library functions,
creating functions,
void functions,
functions that return a value,
Pass by reference
Reading assignment - Chapter 7, 8
Homework assignment
Programming Lab assignment
Problem solving session

Creating data types with Struct

Simple structures,
if else with structures
Repetition with struct
Returning a structure from a function,
Reading assignment - Chapter 10, 11
Homework assignment
Programming Lab assignment
Problem solving session

Arrays

Using arrays,
passing arrays to functions,
Struct with arrays
If else with arrays
Functions with arrays
2-dimensional arrays
strings,
Reading assignment - Chapter 12, 13
Homework assignment
Programming Lab assignment
Problem solving session

Quiz
Days 5&6
Classes

Struct + functions = class
Default Constructors,
Operator overloading,
inheritance
Reading assignment - Chapter 11, 14
Homework assignment
Programming Lab assignment
Problem solving session

Quiz
Days 6&7
More Functions

Function overloading,
recursion,
Reading assignment - Chapters 17
Homework assignment
Programming Lab assignment
Problem solving session

Pointers

pointer,
using pointers with functions,
dynamic memory allocation,
dynamic array allocation,
dynamic string allocation
Reading assignment - Chapters 15
Homework assignment
Programming Lab assignment
Problem solving session

Files and Streams

File input/output,
get and getline,
Reading assignment - Chapter 4
Homework assignment
Programming Lab assignment
Problem solving session

Other Topics - Searching
linear search,
binary search
Sorting
Reading assignment - Chapter 10, 11
Homework assignment
Programming Lab assignment
Problem solving session

Final Exam

In general, a quiz will be scheduled soon after completing the material related to the sections outlined above.

5. Course requirements
   Class participation
   Completion of tests, homework, programming and project assignments neatly on time

6. Tentative Grading policy
You should perform well in all areas to receive an A in this class. In general, an A student has complete understanding of the course material, a B student is strong in many areas, a C student is weak in some concepts, and a D student is weak in many areas.

You are expected to attend all class sessions. While no roll will be taken at each class, a portion of your grade is based on your participation in classroom activities. That means missing classes will affect your grade. There will be no make-up exams during the term. If you miss assignments, and can justify the absences to my satisfaction, I will give you an assignment, probably a programming project, to make up the missed one.

Examinations/quizzes of the type short answer, multiple choice, and fill in the blank questions are designed to test your mastery of terms and concepts, and Essay questions will test your ability to apply those terms and concepts
to problems posed. Two to three in-class exams will be given, along with a final.

Final grades will be based on your performance measured from the following assignments.

- Projects 30%
- Homework 10%
- Tests 60%

Top 15% of the class will receive Grade A
second top 20% of the class will receive a Grade B
third top 30% of the class will receive a Grade C
next 20% of the list will receive a Grade D
rest F

http://www.stcloudstate.edu/stu_handbook/academic_policies/grading.html

7. Project reports - Presentation
The C++ programming project assignments must be completed and presented in report form in a timely fashion. Neatness will be considered as a significant factor. An organized presentation is extremely important in producing a good report. It should be written with titles and subtitles to indicate each section. Tables, lists and programs should be appropriately and completely captioned.

A report should include the following information

- Title page
  - Must bear the project name, date performed and your name
- Problem statement
  - Should have a short statement of the problem, object or purpose
- Input/Output description
- Hand example
- Algorithm development
- Program listing
- Sample test run of the program
- Observations, error handling and general comments
- Conclusions