A. **Academic Division:** Business, Industry, and Technology

B. **Discipline:** Engineering Technology

C. **Course Number and Title:** ENGR1910 Engineering Programming

D. **Course Coordinator:** Lenny Eaken  
   **Assistant Dean:** Daniel Wagner

**Instructor Information:**
- **Name:** Click here to enter text.
- **Office Location:** Click here to enter text.
- **Office Hours:** Click here to enter text.
- **Phone Number:** Click here to enter text.
- **E-Mail Address** Click here to enter text.

E. **Credit Hours:** 3  
   **Lecture:** 2  
   **Laboratory:** 2

F. **Prerequisites:** ENGR1010 and MATH1050

G. **Syllabus Effective Date:** Fall 2017

H. **Textbook(s) Title:**  
   *Introduction to Programming with C++*  
   - **Author:** Y. Daniel Liang  
   - **Copyright Year:** 2013  
   - **Edition:** 3rd  
   - **ISBN # 9780133252811**

I. **Workbook(s) and/or Lab Manual:** None

J. **Course Description:** This course is designed to help students with very little or no computing background learn the basics of building simple interactive applications. The primary method for learning the course material will be to work through multiple “mini-projects”. The student will create technical and simple game programs, debug and test programs, and describe and implement the data structures available. The student will also design a simple algorithm using pseudocode and create programs that implement a variety of common algorithms. Prerequisite: ENGR1010 and MATH1050

K. **College-Wide Learning Outcomes:**

<table>
<thead>
<tr>
<th>College-Wide Learning Outcome</th>
<th>Assessments - - How it is met &amp; When it is met</th>
</tr>
</thead>
<tbody>
<tr>
<td>Communication – Written</td>
<td></td>
</tr>
<tr>
<td>Communication – Speech</td>
<td></td>
</tr>
<tr>
<td>Intercultural Knowledge and Competence</td>
<td></td>
</tr>
</tbody>
</table>

Updated: 2/14/2017
<table>
<thead>
<tr>
<th>College-Wide Learning Outcome</th>
<th>Assessments - - How it is met &amp; When it is met</th>
</tr>
</thead>
<tbody>
<tr>
<td>Critical Thinking</td>
<td></td>
</tr>
<tr>
<td>Information Literacy</td>
<td></td>
</tr>
<tr>
<td>Quantitative Literacy</td>
<td></td>
</tr>
</tbody>
</table>

L. Course Outcomes and Assessment Methods:

Upon successful completion of this course, the student shall:

<table>
<thead>
<tr>
<th>Outcomes</th>
<th>Assessments -- How it is met &amp; When it is met</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Describe computer basics, programs, and operating systems.</td>
<td>Met by performing lab exercises, and responding to quiz questions during weeks 2 &amp; 3.</td>
</tr>
<tr>
<td>2. Write a simple C++ program for console output.</td>
<td>Met by performing lab exercises, and responding to quiz questions during weeks 2 &amp; 3.</td>
</tr>
<tr>
<td>3. Apply the correct syntax and logic to write C++ programs to perform simple calculations using variables to store data.</td>
<td>Met by performing labs, and responding to quiz questions during weeks 4 &amp; 5.</td>
</tr>
<tr>
<td>4. Apply the syntax and logic to declare variables using numeric data types.</td>
<td>Met by performing lab exercises, and responding to quiz questions during week 6 and throughout the remainder of the semester.</td>
</tr>
<tr>
<td>5. Demonstrate use of operators to write numeric expressions.</td>
<td>Met by performing lab exercises, and responding to quiz questions during week 6 and throughout the remainder of the semester.</td>
</tr>
<tr>
<td>6. Convert numbers to a different type using casting.</td>
<td>Met by performing lab exercises, and responding to quiz questions during week 6 and throughout the remainder of the semester.</td>
</tr>
<tr>
<td>7. Demonstrate familiarity with C++ documentation, programming style, and naming conventions.</td>
<td>Met by performing lab exercises, and responding to quiz questions during weeks 6 and throughout the remainder of the semester.</td>
</tr>
<tr>
<td>8. Troubleshoot syntax errors, runtime errors, and logic errors.</td>
<td>Met by performing lab exercises, and responding to quiz questions during weeks 6 through 8.</td>
</tr>
<tr>
<td>9. Apply correctly the syntax and logic to declare type.</td>
<td>Labs and Quizzes during week 8 and throughout the remainder of the semester.</td>
</tr>
<tr>
<td>10. Develop Boolean expressions using comparison operators, implement selection control using one-way, two-way, and nested if statements.</td>
<td>Labs and Quizzes during week 8 and throughout the remainder of the semester.</td>
</tr>
</tbody>
</table>

M. Topical Timeline (Subject to Change):

1. Computers, programs, and C++
   a. Computer basics, programs, and operating systems
   b. The history of C++
   c. C++ program for console output
   d. C++ program-development cycle
   e. Visual C++
   f. Command-line tools on Windows
   g. Command-line tools on UNIX
2. Elementary programming
   a. Perform simple calculations
   b. Input from the keyboard
   c. Identifiers to name elements in the program
   d. Variables to store data
   e. Assignment statements and assignment expressions
f. Constants using the const keyword and #define directive

h. Numeric data types

i. Convert numbers to a different type using casting

j. Char type

k. Program computations

l. C++ documentation, programming style, and naming conventions

m. Syntax errors, runtime errors, and logic errors

n. Debug logic errors

3. Selections
   a. Boolean expressions
   b. Comparison operators
   c. One-way if statements
   d. Two-way if statements
   e. Nested if statements
   f. Common errors in if statements
   g. Selection statements for a variety of examples
   h. Rand function and srand function
   i. Logical operators
   j. Selection statements with combined conditions
   k. Selection control using switch statements
   l. The conditional operator
   m. Stream manipulators
   n. Operator precedence and operator associativity

4. Loops
   a. Executing statements repeatedly using a while loop
   b. Loop with the user confirmation
   c. Loop with a sentinel value
   d. Input redirection
   e. Do-while statements
   f. For statements
   g. Nested loops
   h. Minimizing numerical errors
   i. GCD, Future Tuition, Print Pyramid, Prime Number
   j. Control with break and continue

5. Function basics
   a. Define functions
   b. Returning functions
   c. Void functions
   d. Arguments
   e. Modular code
   f. Ambiguous overloading
   g. Prototypes for function headers
   h. Header files for reusing functions
   i. Implementation
   j. Functions for generating random characters
   k. Applications using the C++ mathematical functions
   l. Applications using the C++ character functions

6. Advanced function features
   a. Arguments by value
   b. Arguments by reference
   c. Pass-by-value and pass-by-reference
   d. Prevent parameters from being modified accidentally
   e. The scope of local and global variables
   f. Runtime efficiency for short functions
   g. Functions with default arguments
h. Functions using stepwise refinement

7. Single-dimensional arrays
   a. Arrays in programming
   b. Declare arrays
   c. Access array elements using indexed variables
   d. Initialize the values in an array
   e. Common array operations
   f. Functions with array arguments
   g. Functions involving array parameters
   h. Search elements
   i. Linear or binary search algorithm
   j. Sort an array using the selection sort
   k. Sort an array using the insertion sort
   l. Process strings using C-strings

8. Multidimensional arrays
   a. Data using two-dimensional arrays
   b. Access array elements in a two-dimensional array
   c. Row and column indexes
   d. Process two-dimensional arrays
   e. Pass two-dimensional arrays to functions
   f. Declare multidimensional arrays

N. Course Assignments:

1. Homework: Selected problems and questions for each chapter must be completed and turned in as homework.
2. Labs: Selected labs will be completed for each chapter throughout the semester
3. Tests: A test will be given at the end of each chapter during the semester.
4. Final: There will be a comprehensive final at the end of the semester.

O. Recommended Grading Scale:

<table>
<thead>
<tr>
<th>NUMERIC</th>
<th>GRADE</th>
<th>POINTS</th>
<th>DEFINITION</th>
</tr>
</thead>
<tbody>
<tr>
<td>93–100</td>
<td>A</td>
<td>4.00</td>
<td>Superior</td>
</tr>
<tr>
<td>90–92</td>
<td>A-</td>
<td>3.67</td>
<td>Superior</td>
</tr>
<tr>
<td>87–89</td>
<td>B+</td>
<td>3.33</td>
<td>Above Average</td>
</tr>
<tr>
<td>83–86</td>
<td>B</td>
<td>3.00</td>
<td>Above Average</td>
</tr>
<tr>
<td>80–82</td>
<td>B-</td>
<td>2.67</td>
<td>Above Average</td>
</tr>
<tr>
<td>77–79</td>
<td>C+</td>
<td>2.33</td>
<td>Average</td>
</tr>
<tr>
<td>73–76</td>
<td>C</td>
<td>2.00</td>
<td>Average</td>
</tr>
<tr>
<td>70–72</td>
<td>C-</td>
<td>1.67</td>
<td>Below Average</td>
</tr>
<tr>
<td>67–69</td>
<td>D+</td>
<td>1.33</td>
<td>Below Average</td>
</tr>
<tr>
<td>63–66</td>
<td>D</td>
<td>1.00</td>
<td>Below Average</td>
</tr>
<tr>
<td>60–62</td>
<td>D-</td>
<td>0.67</td>
<td>Poor</td>
</tr>
<tr>
<td>00–59</td>
<td>F</td>
<td>0.00</td>
<td>Failure</td>
</tr>
</tbody>
</table>

P. Grading and Testing Guidelines:

Click here to enter text.
Q. Examination Policy:
Click here to enter text.

R. Class Attendance and Homework Make-Up Policy:
Click here to enter text.

S. Classroom Expectations:
Click here to enter text.

T. College Procedures/Policies:

Attendance Requirements: All students are required to attend all scheduled classes and examinations. Each faculty member has the right to establish regulations regarding attendance that he/she considers necessary for successful study.

Students who do not attend classes may be administratively withdrawn from those classes. However, failure to attend classes does not constitute withdrawal, and students are expected to process a formal withdrawal through the Student Records Office in Kee Hall.

Student engagement requirements: Student engagement is based on the “active pursuit” of learning which can be measured by class attendance, class participation (in class or online), taking required quizzes/examinations, and submission of work assignments or papers. Student engagement consists of a student attending at least 60% of the class sessions (there should be attendance throughout the term) and/or completing 75% of the assignments listed on the syllabus at the midpoint in the term. Exceptions can be made when there is on-going communication between the student and faculty member. The communication must be documented and the faculty member and student must be in agreement regarding the exception. Students not meeting the expectation will be administratively withdrawn from class. If a student believes he/she was administratively withdrawn in error, he/she may file an appeal. Being administratively withdrawn may have program and financial aid implications.

Academic Misconduct is any activity that tends to compromise the academic integrity of the college, or subvert the educational process. Examples of academic misconduct include, but are not limited to:

1. Violation of course or program rules as contained in the course syllabus or other information provided to the student; violation of program requirements as established by departments and made available to students.

2. Plagiarism including, but not limited to, submitting, without appropriate acknowledgment, any written, visual or oral material that has been copied in whole or in part from the work of others (whether such source is published or not) even if the material is completely paraphrased in one’s own words. This includes another individual’s academic composition, compilation, or other product, or a commercially prepared paper. Plagiarism also includes submitting work in which portions were substantially produced by someone acting as a tutor or editor.

Such practices constitute plagiarism regardless of motive. Those who deny deceitful intent, claim not to have known that the act constituted plagiarism, or maintain that what they did was inadvertent are nevertheless subject to penalties when plagiarism has been confirmed.

3. Cheating and dishonest practices in connection with examinations, papers and projects, including but not limited to using unauthorized notes, study aids or information on an examination; obtaining help from another student during an examination; taking an exam or doing work for another student; providing one’s own work for another student to copy and submit as his/her own; or allowing another
student to do one’s work and then submitting the work as one’s own. Also included would be altering a graded work after it has been returned, then submitting the work for re-grading; or submitting identical or similar papers for credit in more than one course without prior permission from the course instructors.

4. **Fabrication** including but not limited to falsifying or inventing any information, data or citation; presenting data that were not gathered in accordance with defined appropriate guidelines, and failing to include an accurate account of the method by which data were collected.

5. **Obtaining an Unfair Advantage** including, but not limited to stealing, reproducing, circulating, or otherwise gaining access to examination materials prior to the time authorized by the instructor; unauthorized collaborating on an academic assignment; taking, hiding or altering resource material; or undertaking any activity with the purpose of creating or obtaining an unfair advantage over another student’s academic work.

6. **Aiding and Abetting Academic Dishonesty** including, but not limited to providing material, information or other assistance to another person with the knowledge that such aid could be used in any of the violations stated above, or providing false information in connection with any inquiry regarding academic integrity.

7. **Alteration of Grades or Marks** including but not limited to, action by the student in an effort to change the earned credit or grade.

In addition, cases of academic dishonesty may involve photocopied materials. Materials used may fall under the Copyright Act. Violations of said Act may subject the user and/or the College to sanctions.

**Statement on Disabilities:** Any student who requires reasonable accommodations related to a disability should inform the course instructor and the Coordinator of Specialized Services (Room 138 in Kee Hall; phone 419-755-4727).

Students who encounter difficulty in any of their courses are encouraged to visit the Tutoring Resource Center (Room 119 in Fallerius Technical Education Center) for tutoring assistance, and the Student Success Center (Room 136 in Kee Hall) for academic assistance, advising services, referrals for personal counseling and Learning Disability (LD) Testing.

**Statement on Withdrawals:** As a student, you are expected to attend class. If you are unable or choose not to attend class, or if for whatever reason you are unable to keep up with the requirements of a course, you need to officially drop the class at the Student Records Office. Refund dates and withdrawal dates will vary slightly from term to term. Contact the Student Records Office for applicable dates. Additionally these dates are posted on the academic calendar available on the college’s website, [www.ncstatecollege.edu](http://www.ncstatecollege.edu), under the Academics heading on the home page and are available at the Student Records Office in Kee Hall. Students should go to the Student Records Office (Room 142 in Kee Hall) to process their withdrawal from any class.

If you choose to walk away from your class without officially withdrawing from it, the faculty member teaching the class must grade your classroom performance on the material available to him or her. This normally results in an "F" grade. An "F" grade can lower your grade point average considerably depending on the total credits accumulated.