COURSE OUTLINE
Intro to Computer Programming

Course Description
IN252. Introduction to Computer Programming. 3 hours credit. This course will enable the student to develop the skills and knowledge necessary to produce beginning programs and publish to the web. The student will be introduced to programming principles, methodology, style, design, structures, data types, and logic.

Required Materials

* - For complete textbook information, refer to http://www.butlercc.bkstr.com

Supplemental Materials
Flash Drive, 4GB Minimum

Butler-assessed Outcomes
The intention is for the student to be able to:
1. Define the terminology used in programming.
3. Publish developed programs to the web.

Learning PACT Skills that will be developed and documented in this course
Through involvement in this course, the student will develop ability in the following PACT skill area(s):

Analytical Thinking Skills
Critical thinking - Through program development, students will utilize critical thinking skills in design and language.

Personal Development Skills
Personal management – By utilizing the time lines set on various projects, the student will develop time management skills while working on major projects within the course.

Technology Skills
Discipline-specific technology – Through program development and an introductory level of knowledge in programming languages, the student will increase their technology skills.
**Major Summative Assessment Task(s)**

These Butler-assessed Learning Outcome(s) and the Learning PACT skill(s) will be demonstrated by:

1. Completing a portfolio that contains projects from each of the programming languages learned in the course. The portfolio will be electronic and published to the web.

**Skills or Competencies**

Actions that are essential to achieve the course outcomes:

(These skills and competencies are course specific standards)

**Standard 1** – The student will participate in classroom discussion on the history of computers and programming languages.

1. Benchmark 1.1. The student will communicate with other students in peer groups over the history of programming languages and provides valuable insights on the future direction of programming languages.
2. Benchmark 1.2. The student will collaborate with guest speakers and will fully participate in the discussion through active learning.
3. Benchmark 1.3. The student will convert binary numbers to decimal numbers and vice versa.
4. Benchmark 1.4. The student will state the difference between the low-level and high-level programming languages.
5. Benchmark 1.5. The student will draw program diagrams using icons for classes, objects, and other components of object-oriented programming.
6. Benchmark 1.6. The student will name and explain the stages of the software life cycle.

**Standard 2** – The student will develop a simple and small fully functional Java program.

1. Benchmark 2.1. The student will identify the basic components of Java programs.
2. Benchmark 2.2. The student will use the Date, SimpleDateFormat, String, and JOptionPane classes from the standard Java packages.
3. Benchmark 2.3. The student will develop Java programs, using the incremental development approach.
4. Benchmark 2.4. The student will write mathematical expressions using methods in the Math class.
5. Benchmark 2.5. The student will use the GregorianCalendar class in manipulating date information such as year, month, and day.
6. Benchmark 2.6. The student will use the DecimalFormat class to format numerical data.
7. Benchmark 2.7. The student will convert input string values to numerical data.
8. Benchmark 2.8. The student will apply the incremental development technique to writing programs.

**Standard 3** – The student will develop a fully functional C++ program.

1. Benchmark 3.1. The student will utilized several data types including: integers, real numbers, type casting, characters, strings, and user defined data types.
2. Benchmark 3.2. The student will accurately place constants and identified variable declarations in the program.
3. Benchmark 3.3. The student will implement shorthand arithmetic assignment statements.

**Standard 4** – The student will develop a Visual Basic program and/or application.
1. Benchmark 4.1. The student will identify the Main, Form, and Toolbox in the Visual Basic Studio Suite. The student will also set the caption and name property.
2. Benchmark 4.2. The student will add a control to a form and change the property for more than one control at a time.
3. Benchmark 4.3. The student will add an image control and/or scroll bar to a form with a command button.
4. Benchmark 4.4. The student will create an Object Oriented Event Driven application.
5. Benchmark 4.5. The student will write a Visual Basic equation including the Val function and Format function.

**Standard 5** – The student will design, create, and develop a five-page website utilizing HTML and cascading style sheets.
1. Benchmark 5.1. The student will write five pages of HTML that includes the basic tag structures H1, H2, H3, UL, OL, font, and other basic elements and attributes.
2. Benchmark 5.2. The student will create an external style sheet and link to the current HTML files.
3. Benchmark 5.3. The student will publish the website and check browser compatibility as well as error-checking.

**Standard 6** – The student will develop a published electronic portfolio that showcases each of the programs and website projects mastered in the course.
2. Benchmark 6.2. The student will complete a peer-reviewed evaluation of other electronic portfolios and given recommendations for improvement as well as suggestions for future developments.

**Learning Units**
I. Overview of computers and programming
   A. History of programming languages
   B. History of computers
   C. Low-level and high-level computer languages
   D. Program diagrams for object-oriented programming

II. Java
   A. Basic components of Java programs
   B. Mathematical expressions
C. Different class structures
   1. Gregorian calendar
   2. Decimal format
D. Strings
E. Incremental development technique

III. C++
A. Data types
   1. Integers
   2. Real numbers
   3. Type casting
   4. Characters
   5. Strings
   6. User defined data types
B. Constants
C. Variable declarations
D. Shorthand arithmetic assignment statements

IV. Visual Basic
A. Visual Basic suite
B. Object oriented events
C. Equations
   1. Val function
   2. Format function

V. HTML and Cascading Style Sheets
A. Nesting and non-nesting
B. Basic tag structure including attributes and variables
C. Tables
D. Lists
E. Images
F. Hyperlinks
G. Basics of Cascading Style Sheets
H. External and Internal Cascading Style Sheets
I. Publish website to server

VI. E-Portfolios
A. Importance of developing an e-portfolio
B. Peer-review

Learning Activities
The student will engage in classroom discussion, participate in guest speaker panel expos, communicate with online students through CISCO Telepresence, and participate in portfolio presentations. Instruction will be in the form of lecture, video, group work, and modeling.
Grade Determination
The student will be graded on learning activities and assessment tasks. Grade determinants may include the following: daily work, quizzes, chapter or unit tests, comprehensive examinations, projects, presentations, class participation, and other methods of evaluation at the discretion of the instructor.