Butler County Community College
Business & Industrial Technology Division

Bob Ray & Roger Dunnell
Fall 2003

COURSE OUTLINE
Introduction to Programming

Course Description: IN 105. Introduction to Programming. 3 hours credit. This course will enable the student to solve algorithmic problems and logically design a computer program using structured programming techniques. The student will also learn the most commonly used Information Technology terms. The course will enable the student to apply these skills to application programming or Web application development. Before enrolling in the course, the student should be competent in basic computer operation.

Course Relevance: Private and public organizations have a need for computer program development skills, usually using multiple programming languages, to improve productivity, increase efficiency and moderate costs. This course provides the basic concepts which will allow the student to develop the skills necessary to understand create and modify customized application programs. These hardware and software concepts are the foundation behind all modern computer systems. They will help the student understand how computers, computer programs and computer information systems can be used to solve a given business problem. Businesses are eagerly looking to hire professionals with these programming skills; therefore, the skills developed in this class are directly transferable to any business or professional situation.

Required Materials:
Farrell, J. (2002). *Programming Logic and Design* (2nd ed.). Boston, MA: Course Technology. (ISBN: 0-619-06315-7)

Supplemental Materials:
USB Digital Drive: 64 MB minimum

Learning Outcomes:
The intention is for the student to be able to:
1. Articulate and demonstrate a basic understanding of the fundamental concepts of structured program development
2. Articulate and demonstrate a basic understanding of fundamental information technology vocabulary

Primary Learning PACT Skills that will be DEVELOPED and/or documented in this course:
Through the student involvement in this course, he/she will develop his/her ability in the following PACT skill areas:

1. Problem solving
   - Through problem solving exercises, the student will develop the skill to identify and resolve technical problems.
2. Critical thinking
   - Through the inception, planning, design and implementation of individual and team projects, the student will develop a process for critical thinking.
3. Field-related technology
   - By developing examples, exercises, and applications related to specific fields, the student will improve understanding of his/her own field or other technological or vocational field.
4. Computer literacy
   - The student will use a computer to develop and practice his/her programming skills.

Secondary skills (developed but not documented):

Reading
Writing
Listening
Internet use
Teamwork

Major Summative Assessment Task(s):

These learning outcomes and the primary Learning PACT skills will be demonstrated by:

1. Completing a team project to demonstrate a fundamental understanding of programming concepts.
2. Completing an individual project to apply fundamental programming concepts to a personal or work related problem or complete a final exam covering fundamental programming concepts.

Course Content:

I. Themes - Key recurring concepts that run throughout the course:
   A. Hardware and software elements of modern information systems
   B. Problem discovery, analysis and documentation process
   C. Commonality of program design, development and implementation methodologies
   D. Language specific syntax implementation of structured programming, variable declaration, block structured and modular programming concepts

II. Issues - Key issues that will be addressed in this course: areas of conflict that must be understood in order to achieve the intended outcome:
   A. Understanding the inherent hardware and software limitations and their impact on application development

III. Concepts - Key Concepts that must be understood to address the issues:
   A. The common techniques for translating problem solutions into computer

programs
B. Importance of algorithmic thinking in problem analysis and program development
C. Evolution and purpose of specific programming languages
D. The role of teams and individuals in project development

V. Skills/Competencies - Actions that are essential to achieve the course outcomes:
A. Describe the relationship and interaction of the hardware and software components of an information system
B. Use the Internet to find and price the components of a simple office information system
C. Describe the use of logic in the programming process
D. Write a simple computer program
E. Demonstrate the use of looping
F. Use AND/OR logic to demonstrate programming decision making
G. Demonstrate an understanding of arrays
H. Articulate the use of files and records in sequential files
I. Articulate the concept of object-oriented programming

Learning Units:

I. Introduction to Computers and Logic

II. Introduction to Modules and Documentation

III. Development of a Simple Program

IV. Basic Decision Structures

V. Basic Repetition Structure

VI. Introduction to the C Language

VII. Basic Array Concepts

VIII. Introduction to the JavaScript Language

IX. Basic File Processing Concepts

X. Introduction to the Java Language

XI. Introduction to Visual Basic

Learning Activities:

Independent and collaborative learning activities will be assigned within and outside the college classroom to assist the student in achieving the intended

learning outcomes. Learning activities will involve the student in the creation and design of applications independently or in collaboration with others. This will include various aspects of design, technology, and project management. Homework, small group projects, readings, computer exercises, and research assignments will prepare the student to be equipped to successfully complete the major summative assessment tasks.

Grade Determination:
The student will be graded on satisfactory completion of major summative assessment tasks, attendance, participation and the timely completion of class exercises and tutorials.