

COFFEYVILLE COMMUNITY COLLEGE

**COMP-123
COURSE SYLLABUS
FOR
INTRODUCTION TO COMPUTER PROGRAMMING: BASIC
FALL 2004**

**KENDALL PAYNE
INSTRUCTOR
MATH SCIENCE DIVISION**

COURSE #: COMP-123

COURSE TITLE: Introduction to Computer Programming: BASIC

CLASS TIME: 9:10 – 10:10 AM MWF

CREDIT HOURS: 3 Credit Hours

INSTRUCTOR: Kendall Payne

OFFICE LOCATION: Room 119, Weinberg Hall

OFFICE HOURS: Posted on Office Door

TELEPHONE: 251-7700 Ext. 2126

E-MAIL: kendallp@coffeyville.edu
kpayne12@cox.net
E-mail both addresses

PREREQUISITES: None

REQUIRED TEXT AND MATERIALS: Fundamentals of QBASIC Programming
Robert C. Nickerson
Harper-Collins College Publishers, 1995

COURSE DESCRIPTION: This course introduces the beginning student to simple computing algorithms, writing pseudocode, and drawing flowcharts. The BASIC language is used and topics covered up to string manipulation are included. Extensive lab experience is provided in this course.

EXPECTED LEARNER OUTCOMES

1. Become familiar with basic PC and floppy disk commands and operations.
2. Learn to utilize the standard QBASIC editor to create, edit, print and save programs written in QBASIC.
3. Learn and incorporate the nine essential QBASIC statements in writing programs.
4. Learn to create and properly "type" variables and "constants".
5. Learn to properly display program output to both and screen and printer.
6. Learn to write pseudocode for algorithms and be able to convert the algorithms to machine executable computer programs.
7. Learn to draw flowcharts of computer algorithms and programs using the six basic flowchart symbols.
8. Learn to create and utilize the "IF-THEN-ELSE" statement in program decision making.
9. Learn to create and utilize the "Select Case" decision within the general program coding.
10. Learn to create and use program iterations(looping) in program design and coding.
11. Learn to incorporate the concepts of "stepwise refinement" and "Topdown design" in QBASIC programming.
12. Learn to incorporate the use of typical "scoping" of variables.

13. Learn to create and utilize sub-procedures and functions in program design and coding.
14. Learn to recognize, create, and manipulate "string" variables in program code writing
15. Learn to create and utilize user-friendly, attractive user input/output screens.
16. Learn to create and use one-dimensional arrays in programming.
17. Learn to incorporate and utilize two-dimensional arrays within general program coding.
18. Learn to access and use a sequential data file.

LEARNING TASKS AND ACTIVITIES

- 8/16 Introduction, QBASIC commands and editor
- 8/18 Entering and running a simple BASIC program
- 8/20 Declaring and naming variables and constants
- 8/23 BASIC arithmetic operations
- 8/25 Order of precedence for math operators
- 8/30 Designing banners
- 9/1 MOD, DIV, INT () and ^ operators
- 9/3 Logical operators
- 9/6 **LABOR DAY – NO CLASS**
- 9/8 Output formats
- 9/13 **EXAMINATION #1**
- 9/15 Control structures
- 9/17 Pseudocode and algorithms
- 9/20 Flowcharts
- 9/22 IF-THEN and IF-THEN-ELSE statements
- 9/27 SELECT-CASE structure
- 9/29 FOR NEXT loops
- 10/1 WHILE WEND loops
- 10/4 DO loops
- 10/6 **EXAMINATION #2**
- 10/11 Step wise refinement
- 10/13 Top-down design
- 10/15 Scope of variables
- 10/18 Subroutines
- 10/20 Functions
- 10/25 Strings
- 10/27 String functions
- 10/29 Designing input and output screens
- 11/1 More on designing input and output screens
- 11/3 **EXAMINATION #3**
- 11/8 PRINT USING statement
- 11/10 One-dimensional arrays
- 11/12 Two-dimensional arrays
- 11/15 Linear search algorithms
- 11/17 Sorting
- 11/22 Data files
- 11/29 Reading and writing data files
- 12/1 Sequential and random data files
- 12/3 **EXAMINATION #4**
- 12/6 Colors
- 12/9 **FINAL EXAM**

ASSESSMENT OF OUTCOMES

The student will be assessed in three areas:

A. Cognitive:

Knowledge and understanding of the materials. Knowledge of all areas of material will be assessed through exams which are mainly objective in nature (Multiple Choice and Matching questions), with additional short answer/essay questions. (30% of grade)

B. Metacognition:

Each student will be required to show how they can incorporate the Cognitive aspects of this material attained from the text and lectures by answering study guide questions. These questions will represent the different levels of learning. These will be presented in written and verbal form. (20% of grade)

C. Affective

Attendance, attitude, assignments and participation in classroom discussion and exercises. (20% of grade)

D. Performance and skills

The performance and skills learned in this course will be demonstrated and evaluated in the computer lab (30% of grade)

GRADING POLICY

Semester grades will be based upon the following:

1. Unit tests
2. Computer Programs
3. Pop quizzes
4. Homework
5. Final exam

UNIT TESTS

There will be four unit tests. Each test will be worth 100 points. As a student you are required to be present for all exams. If you cannot be present for an exam, you must notify me IN ADVANCE. If you are not present for an exam, and I have not heard from you by the day of the exam, you will not be allowed to make up the exam. A '0' will be placed by your name in the grade book for that exam. The dates of the exams are listed on the next page.

LAB TIME

Students may need to work extra in the lab in order to complete assignments. In order to learn how to use the computer and software, a student must spend time in the lab.

POP QUIZZES

Occasionally, I will give pop quizzes at the beginning of class. Each quiz will be worth 10 to 20 points. These quizzes will be unannounced. If you miss class that day with an unexcused absence, you will not be allowed to make up the quiz.

HOMEWORK

You will be given homework to do nearly every class period. The homework will consist of both reading and written assignments. On some days, I will pick up the homework. On the other days, we will go over the homework in class. Homework exercises will be 20 to 40 points apiece. Each assignment will have a due date. Each assignment must be turned in at the beginning of class on the day that it is due. Any assignment not turned in at the beginning of class **WILL NOT BE ACCEPTED**.

LAB HOURS

Will be announced in class

REMINDER: DO NOT PRINT WHILE AN INSTRUCTOR IS LECTURING

COMPUTER PROGRAMS

You will be given computer programs to write nearly every class period. These will be done on the personal computers in the lab(Room 104). The programs will be written to implement different applications in QBASIC. Each program will be worth 10 to 30 points depending on the size of the assignment and the level of difficulty. Each assignment is due at the beginning of class on the day that it is due. Any assignment not turned in at the beginning of class will not be accepted.

FLOPPY DISKS Each student will be given a 3 ½ inch floppy disk to used for computer projects in this class. The disk must stay in Room 104 -- Occupational Building(Computer Lab).

FINAL EXAM The final exam will be comprehensive and will be worth 100 points. Your final exam is on Thursday, December 9, 2004 from 10:00 – 11:40 AM. All students must take the final exam on this date at this time. The final will not be given at any other time. **NO EXCEPTIONS!!**

GRADING SCALE

A	90 – 100%
B	80 – 89%
C	70 – 79%
D	60 – 69%
F	0 – 59%

4 exams (@100 points)	400 points
Computer programs	300 points
Homework/pop quizzes	200 points
<u>Final exam</u>	<u>100 points</u>
TOTAL POINTS	1000 points

EXAM DATES:

Examination 1	September 13, 2004
Examination 2	October 6, 2004
Examination 3	November 3, 2004
Examination 4	December 3, 2004

Final ExamDecember 9, 2004

INCOMPLETES: Incomplete grades for the semester will be given in case of emergencies and only by mutual consent of the student and the instructor.

ATTENDANCE: Each student is required to attend every class session. Only in the event of illness or an emergency will you be excused from class. All other absences will be classified as unexcused absences. In event of illness or emergency, you must notify me personally. My phone number is 251-7700, Ext. 2126. If you are not in class and I have not heard from you by the end of that day, you will be given an unexcused absence.

A summary of excused and unexcused absences is listed below:

EXCUSED ABSENCES:

- Illness
- Emergency(Personal or family related)
- Participation in a school related activity or sporting event

For those students that have to miss class due to school related activities(sports, music, etc), these absences will not count toward the three excused absences provided that their exams and/or homework are made up **prior** to missing class.

NOTE: Each student is allowed only three excused absences. After the third excused absence, all absences become unexcused absences.

For excused absences, it is your responsibility to get in touch with me to make up any tests and/or homework. Any tests and/or homework that need to be made up must be done by the next class period. After the second excused absence, no tests and/or homework can be made up.

Those students that must miss class because of a school related activity must make up any exams and homework they will miss before the day they are going to miss class.

The instructor must validate all excused absences.

UNEXCUSED ABSENCES:

- All other absences

For unexcused absences, you will not be allowed to make up the work that you missed. **THIS INCLUDES EXAMS.**

Competencies for INTRODUCTION TO COMPUTER PROGRAMMING: BASIC

1. The student will be familiar with basic PC and floppy disk commands and operations.

- A. Properly use the PC and network commands.
- B. Identify the various disk drives and insert floppy disks properly.
- C. Be able to save and retrieve files to and from various disk drives.

2. The student will learn to utilize the standard QBASIC editor to create, edit, print, and save program written QBASIC.

- A. Properly enter and exit QBASIC editor.
- B. Use the drop-down menu system of the QBASIC editor to facilitate the writing, running, and debugging of program code.
- C. Open, close, save, and print files to the printer via the QBASIC editor.

3. The student will memorize and be able to incorporate the nine essential BASIC statements in their code writing.

- A. Know the basic statements: CLS, REM, PRINT, LET, GOTO, IF-THEN, END, READ-DATA, INPUT.
- B. Enter and run a simple program.
- C. Design and enter a typical program banner.
- D. Utilize correctly the commands KILL, NAME, NEW, and SYSTEM.

4. The student will learn to create and properly "type" variables and "constants".

- A. Know how to declare, name, and type basic VARIABLES and CONSTANTS.
- B. State the difference between numeric & string variables.
- C. Know and properly utilize the "Order of Precedence" of math operators.
- D. Correctly identify the use and results of operators such as MOD, INT, DIV, (), and ^.
- E. Know and use correctly the logical operators AND, OR, and NOT.

5. The student will be able to properly display program output to both screen and hard copy.

- A. Utilize properly the "comma" in print statements.
- B. Utilize properly the "semicolon" in print statements.
- C. Utilize properly the "TAB" in print statements.

6. The student will be able to write pseudocode for simply programming algorithms and programs.

- A. State and provide an example of the three major "CONTROL" structures found in programming languages.
- B. Write "pseudocode" for a given computer algorithm.

7. The student will learn to draw accurate flow charts of computer algorithms and programs using the basic six flowcharting symbols.

- A. Read correctly a flowchart associated with a simple computer program.
- B. Know the meaning of the "start/stop oval", "process box", "I/O slant box", "decision diamond", "connector circle", "logic arrow".
- C. Draw an accurate "FLOW-CHART" for a given computer algorithm using the six basic symbols.

8. The student will learn to create and utilize the "IF-THEN-ELSE" statement in program decision making.

- A. Properly utilize the IF-THEN-ELSE in horizontal form.
- B. Properly utilize the IF-THEN-ELSE in block form.
- C. Properly utilize more than one IF-THEN statements in a "nested" configuration.

9. The student will learn to create and utilize the "Select-Case" decision within general program coding.

- A. Know and use the various "relational operators" in a SELECT-CASE structure.
- B. Utilize the "CASE ELSE" option of the SELECT-CASE structure.
- C. Properly utilize the "SELECT CASE" statement in a "multiple" IF-THEN capacity.

10. The student will learn to create and use program iterations (looping) in program design and coding.

- A. Identify the five basic steps to looping process in computer programming.
- B. State the difference between conditional and unconditional loops.
- C. Compare and use FOR-NEXT, LOGIC, WHILE-WEND, and DO LOOPS.

11. The student will learn to incorporate the concepts of proper program design in the coding.

- A. Define and use the concept of "stepwise refinement" in writing program code.
- B. Define and use the concept of "Topdown design" in designing and writing computer code.

12. The student will learn to incorporate use the typical "scoping" of variables.

- A. Define and properly utilize "Local" variables.
- B. Define and properly utilize "Global" variables.
- C. Describe the function of the statements, "common shared".

13. The student will learn to create and utilize Sub-procedures and Functions in program design and coding.

- A. Effectively utilize the concepts of SUB-PROCEDURES and FUNCTIONS in program design and coding.
- B. State the difference between SUB-PROCEDURES and FUNCTIONS.

14. The student will learn to recognize, create manipulate "String" variables in program code writing.

- A. Make string comparisons successfully.
- B. Build and concatenate strings.
- C. Define the term "palindrome".
- D. Know and be able to use the statements, MID\$, LEN, LEFT\$, RIGHT\$, INSTR, VAL, CHR\$, INKEY\$, STRING\$, ASC, STR\$.

15. The student will learn to create and utilize user-friendly, attractive user input/output screens.

- A. Utilize the statements, COLOR, LOCATE, SLEEP in code writing.
- B. Utilize properly the statement PRINT USING to format output in a professional manner.
- C. When shown a PRINT USING statement describe the correct output.
- D. Utilize PRINT USING and LPRINT to produce professional looking printed reports.

16. The student will learn to create and use one-dimensional arrays in programming.

- A. Create and utilize one-dimensional arrays in program code.
- B. Discuss and be able to code the three cases of linear search algorithms.
- C. Discuss and be able to code a simple sort algorithm.

17. The student will learn to create and utilize two-dimensional arrays within general program coding.

- A. Create and load data into a two-dimensional array (table).
- B. Use two-dimensional arrays for data lookup.

18. The student will learn to access and use a sequential data file.

- A. Describe the use and advantages of disk data files.
- B. Describe the differences between "sequential" and "random access" data files.
- C. Create and recall a sequential data file
- D. Recall and update a sequentially stored data file.