

JOLIET JUNIOR COLLEGE
DEPARTMENT OF COMPUTER INFORMATION
AND OFFICE SYSTEMS

COURSE SYLLABUS

Course Prefix and Number	CIS 130
Course Title	BASIC Programming
Curriculum	Computer Information Systems
Semester Hours	4
Lecture	4
Lab	0
Prerequisites	CIS 122 and Math 094 or equivalents, or Consent of Department

Catalog Description

An introduction to computer programming in the BASIC (Beginners All Purpose Symbolic Instruction Code) language. This course is designed to teach problem analysis, program design and programming in the QBASIC language. The student will use a PC-based programming environment. An introduction to Visual Basic for Windows is included.

Course Objectives: See attached.

Prepared by:

Reviewed by:

Tim Vanderwall
Dept. of CIOS Updated 12/00
 Revised 10/98
 Revised 7/97
 Revised 2/92
 Revised 11/91
 Revised 11/89

Ram Raghuraman
Department Chairperson Date

STUDENT MATERIALS

A. Textbooks:

Title: Quick Basic & QBASIC Using Modular Structure, w/ Visual Basic

Author: Bradley

Publisher: Irwin

B. Other Required Materials

Two 3-1/2" high density diskettes

C. Other Course Requirements

The student will:

1. Operate a microcomputer using Microsoft QBASIC software and Microsoft Visual Basic software.
2. Create, list, execute and debug programs using DOS and Windows environments. Identify and use flowchart techniques and/or pseudocode to illustrate the logical flow of problem solving. Analyze problems, design the algorithm for solution. Code in BASIC the algorithms, making use of arithmetic and relational expressions, branching techniques, I/O commands, loops, function usage, and one and two-dimensional array allocation commands. Specify what is being accomplished in the program and why via proper internal documentation. Output should be realistically documented and formatted. Debug, reanalyze and recode program segments that do not accomplish desired results. Sort lists using various sort routines. Optimize program segments whenever possible. Use sequential and random DATA files for I/O. Make use of the RND (Random Number Generator) function of BASIC in writing simulations. Use the Visual Basic supplement to explore the Microsoft Visual Basic programming environment.

Student Evaluation (Type of Grading)

The student will be evaluated on theory by examinations and on practice by programming assignments.

<u>Week/Days</u>	<u>Topic or Class Activity</u>	<u>Teaching Aids or Special Instruction</u>
1	General computer systems concepts; Hardware and software; Program Planning; Using the QBASIC environment; Flowcharting and pseudocode	
2	BASIC statements and program format; Modular programs with calculations and strings; Variables and constants; Elements of BASIC, orders of operations; Subroutines; INPUT-LET-PRINT	
3	Programs with Loops; DO-LOOP; Counters and accumulators; Output formatting; PRINT USING	
4	Selection structures; IF-THEN, IF-THEN-ELSE; READ/DATA-RESTORE	
5	Structured programming concepts; Control structures; Internal documentation; Top-Down Programming and Testing	
6	Report design and subtotals; Control break logic	
7	Data validation; Interactive programming; String functions	
8	Text graphics and music; Color graphics concepts	
9	Menus; Select Case statement	
10	Numeric functions; User defined functions Random number generation; FOR-NEXT loops Introduction to functions and subprograms	
11	Single-dimension arrays; Searching arrays	
12	Two-dimension arrays; Sorting	
13	External file- -transaction processing; Sequential files	
14	Random files	
15	Visual Basic concepts (supplement)	

OBJECTIVES

At the end of the course, the student will be able to:

1. Identify basic computer systems concepts;
2. Plan a problem for programming;
3. Create a flowchart and write pseudocode for a programming problem in BASIC;
4. Identify and utilize BASIC command statements with appropriate programming formats, such as INPUT-LET-PRINT, DO-LOOP, PRINT USING, IF-THEN, IF-THEN-ELSE, READ/DATA-RESTORE, SELECT CASE, AND FOR-NEXT;
5. Identify and utilize order of operations within programming statements;
6. Create and utilize subroutines;
7. Create and utilize counters and accumulators;
8. Perform appropriate output formatting;
9. Utilize structured programming concepts and identify control structures;
10. Create internal and external program documentation;
11. Utilize top-down programming concepts;
12. Perform program testing and validation;
13. Utilize control break logic within programs;
14. Perform data validation;
15. Incorporate text or color graphics and music into a program;
16. Create program menus;
17. Utilize numeric and user-defined functions and perform random number generation;
18. Create, search and sort arrays;
19. Perform external file-transaction processing on sequential and random data files.
20. Use subprograms and functions in a program.
21. Create a Visual Basic project so as to compare and contrast the QBASIC environment with the Visual Basic windows environment.