

COURSE OUTLINE

Computer Science Information Systems 130 Introduction to Algorithms

Catalog Statement

CS/IS 130 is a course in programming, algorithm development and problem-solving using both object-oriented and structured approaches. It includes a study of syntax and data structures with applications in science, engineering, and industry. This course is suitable either for students planning to transfer or those wishing to develop a marketable skill.

Total Lecture Units: 3.0

Total Laboratory Units: 0.0

Total Course Units: 3.0

Total Lecture Hours: 48.0

Total Laboratory Hours: 0.0

Total Laboratory Hours To Be Arranged: 0.0

Total Faculty Contact Hours: 48.0

Prerequisites: CS/IS 112 or equivalent.

Course Entry Expectations

Prior to enrolling in the course, the student should be able to:

- examine problems, think in a logical fashion, and provide solutions/algorithms for the problems;
- show the solution/algorithm using flowcharts or pseudocode;
- utilize a compiler to write, debug, and test Java programs.

Course Exit Standards

Upon successful completion of the required coursework, the student will be able to:

- write simple programs using basic unstructured data types;
- use procedures for problem-solving and modularity;
- use objects for data encapsulation;
- develop algorithms and select an appropriate combination of algorithm and data structures for various problems;
- write more complex programs using structured data types, objects, parameters, functions and recursion;

- describe and perform basic operations with reference variables, linked lists, and binary trees;
- convert simple documentation into basic programming including looping and decision;
- refine a program into two or more modules.

Course Content

Total Faculty Contact Hours = 48.0

Introduction (4 hours)

- Problem definition
- Object concepts

Basic Concepts (3 hours)

- Data types
- Scalar data types
 - Integer types
 - Floating point types
 - Character

Elementary Programs (3 hours)

- Elementary programs
- The assignment statement
- Program examples

Control Flow (3 hours)

- Looping statements
- Conditional statements
- Selection statements

Running, Debugging, and Testing Programs (6 hours)

- Debugging
- Program testing program
- Documentation and maintenance

Algorithm Development (5 hours)

Additional Data Types (6 hours)

- User - defined
- Arrays
- Files

Functions and Methods (5 hours)

- Declaring and invoking a function
- Local and global variables
- Parameters – by reference and by value
- Recursion

Objects (8 hours)

- Declaring classes
- Inheritance
- Arrays of objects
- Files of objects

References and Linked Lists (5 hours)

Reference variables
Stacks and queues
Linked lists
Binary trees

Methods of Instruction

The following methods of instruction may be used in the course:

- lecture;
- hands-on programming problems using computers.

Out of Class Assignments

The following out of class assignments may be used in the course:

- programming assignments (e.g. write a simple program incorporating objects, functions and methods).

Methods of Evaluation

The following methods of evaluation may be used in the course:

- quizzes;
- midterm examinations;
- programming projects;
- final examination.

Textbook(s)

Gaddis, Tony. *Starting Out with JAVA: From Control Structures through Objects* 6th ed. Upper Saddle River: Prentice Hall, 2015. Print.

11th grade Textbook Reading Level. ISBN: 978-0133957051

Student Learning Outcomes

Upon successful completion of the required coursework, the student will be able to:

- analyze problems and provide logical solutions;
- outline solutions in a systematic fashion;
- code, test, and debug intermediate level Java programs.