



South Central College

COMP 1120 Foundations of Computing

Course Outcome Summary

Course Information

Description This course introduces the student to the world of information systems and technology. Students will explore the history of computing, career opportunities in information technology, computer concepts as they apply to a business environment, basic web page development, command-line interfaces, file management principles, computer numbering systems, and database principles. Students will also receive initial exposure to computer programming and algorithms through the use of problem analysis, pseudo-code and entry-level programming. (Prerequisites: None)

Total Credits 4

Total Hours 64

Types of Instruction

Instruction Type	Credits/Hours
Lecture	4/64

Pre/Corequisites

Prerequisite None

Institutional Core Competencies

Civic Engagement and Social Responsibility - Students will be able to demonstrate the ability to engage in the social responsibilities expected of a community member.

Critical and Creative Thinking - Students will be able to demonstrate purposeful thinking with the goal of using a creative process for developing and building upon ideas and/or the goal of using a critical process for the analyzing and evaluating of ideas.

Course Competencies

1. Describe the history of computing.

Learning Objectives

Explore the history of computing.

Identify turning-point events in the technology era.

2. Summarize and compare the employment prospects in the information technology field.

Learning Objectives

Describe the organization of an information technology department.
Explore, categorize and assess career options available to the IT professional.

3. Identify hardware and software platforms used in business.

Learning Objectives

Describe various hardware platforms and technologies used throughout an organization.
Discuss enterprise computing.
Explore the use of emerging technologies in business.
List various software categories found in an organization.
Differentiate between operating systems and applications software.
Discuss issues of software licenses and open source software.

4. Apply file management principles on Windows system, manipulating file and folder structures.

Learning Objectives

Utilize both the graphical user interface (GUI) and command-line interface (CLI).
Navigate among directories in a file system.
Create a folder structure.
Copy, move, delete, and rename files and folders.

5. Apply file management principles on a Unix-like system, manipulating file and folder structures.

Learning Objectives

Copy files using the copy (cp) command.
Move and rename files using the move (mv) command.
Remove files using the remove (rm) command.
Create and remove directories using common Linux utility programs and shell commands.
View and manipulate the attributes of files and directories.
Connect to and work with files on a Linux-based server using various utilities.
Use editors to create and edit standard Linux data files.

6. Work with numbering systems.

Learning Objectives

Explain the binary representation of data in ASCII.
Perform conversion between numbering systems (binary, octal, and hexadecimal).

7. Examine data management in information systems.

Learning Objectives

Discuss the hierarchy of data.
Compare database management systems.
Describe the use of data warehouses and data mining in the organization.

8. Explore ethical issues in computing.

Learning Objectives

Discuss issues related to data privacy.
Explain methods to protect against identity and information theft.
Discuss issues related to computer crime.

9. Explain the concepts of networking, the Internet, and the World Wide Web.

Learning Objectives

List and describe the basic networking concepts.
Describe and contrast the concepts of the Internet and the Web.
Investigate standard concepts associated with the Internet, security, and the Web.
Create a basic web page using Hypertext Markup Language (HTML) text editor.

10. Compare types and usage of common programming languages.

Learning Objectives

Summarize the evolution of programming languages.
Identify distinguishing characteristics of those programming languages in common use today.
Compare the advantages and disadvantages of the main programming paradigms (e.g. procedural, object-oriented, functional).

11. Employ algorithms to solve problems.

Learning Objectives

Discuss why algorithms are useful in problem-solving.
List the recommended steps in solving a problem.
Create algorithms for solving simple problems.
Use pseudo-code to implement an algorithm.
Analyze an algorithm's correctness and efficiency.

12. Explain machine architecture and language.

Learning Objectives

Illustrate the architecture of a computer.
Differentiate between high-level languages and machine language.
Write a machine language program and execute it on a simple simulator.

13. Discuss syntax and semantics of common programming languages.

Learning Objectives

Explain the use of variables, types, expressions, and assignments.
List the primary conditional and iterative control structures.
Choose appropriate programming constructs for a given task.
Write and test a program that implements a simple algorithm.

14. Employ event-driven programming techniques.

Learning Objectives

Explain the concept of event-driven programming.
Identify which programming languages in common use today support events.
Write and test a simple program that reacts to simple user-generated events.

SCC Accessibility Statement

South Central College strives to make all learning experiences as accessible as possible. If you have a disability and need accommodations for access to this class, contact the Academic Support Center to request and discuss accommodations. North Mankato: Room B-132, (507) 389-7222; Faribault: Room A-116, (507) 332-7222.

Additional information and forms can be found at: www.southcentral.edu/disability

This material can be made available in alternative formats by contacting the Academic Support Center at 507-389-7222.