



BAKER COLLEGE

STUDENT LEARNING OUTCOMES

CS3210 Data Structures and Algorithms I
3 Semester Credit Hours

Student Learning Outcomes and Enabling Objectives

1. Implement algorithms for solving problems using Python.
 - a. Learn software algorithm analysis
 - b. Use the proper notation while writing software programs
 - c. Develop proper logic to write smart programs
2. Use principles of recursion to solve problems.
 - a. Explain principles of recursion
 - b. Choose the proper recursion principle to solve a particular problem
3. Develop python code to implement stacks.
 - a. Identify the operations of the ADT stack.
 - b. Implement a program that utilizes the stack ADT.
 - c. Evaluate the benefits associated with Array-Based stack implementations.
4. Create an application that utilizes appropriate discrete data structures to solve business problems.
 - a. Apply the principles of encapsulation in your programs
 - b. Apply the principles of Inheritance in your programs
 - c. Apply the principles of Polymorphism in your programs
5. Identify abstract data types that can be used in software development.
 - a. Apply basic software engineering principles and algorithms in writing professional programs to solve problems
 - b. Describe an abstract data type (ADT) in Python.
 - c. Describe an ADT bag object.
6. Develop an application that makes use of abstract data types.
 - a. Apply different data types in your programs
 - b. Use array logic in developing software programs
 - c. Implement simple and structured data types as arrays

Big Ideas and Essential Questions

Big Ideas

- Apply basic software engineering principles
- Develop and implement simple data types and structured data types
- Apply principles of Object-Oriented Development and Programming

- Demonstrate the ability to use principles of recursion to solve problems.
- Implementation of advanced programming techniques

Essential Questions

1. How to apply basic software engineering principles and algorithms in writing professional programs to solve problems?
2. How to use arrays and apply their logic in developing software programs?
3. How to apply the principles of encapsulation, inheritance, and polymorphism including function and operating overloading to improve program functionality?
4. How to use principles of recursion to solve problems?
5. How to implement stacks?
6. How to research, design, develop, test, run, debug, and implement an advanced-level python program?

These SLOs are approved for experiential credit.

Effective: Spring 2023