

CSCI 136 – Fundamentals of Computer Science II
Exam III Study Outline

I. Previous Concepts

- A. Linked Lists
- B. Stacks and Queues
- C. Abstract Data Types and Generics
- D. Performance
- E. Inheritance (Python)
- F. Recursion
- G. Threads / Concurrency
- H. Linux

II. C++ Basics

- A. Program Structure
 - 1. Preprocessor Directives
 - 2. main Function
 - 3. Compilation and Execution on Linux
 - 4. Comments
 - 5. Namespaces
- B. Variables and Data Types
 - 1. Variable Names
 - 2. Fundamental Data Types
 - 3. Declaration and Initialization
- C. Operators
 - 1. Assignment
 - 2. Arithmetic
 - 3. Compound
 - 4. Increment/Decrement
 - 5. Relational
 - 6. Logical
 - 7. Ternary Operator
 - 8. Bitwise Operators
 - 9. Type Casting
- D. Basic Input/Output (I/O)
 - 1. Input/Output Streams
 - 2. Input Conversion
- E. Control Structures
 - 1. Conditional
 - 2. Selection (Switch)
 - 3. Iteration
 - a. while Loop
 - b. do-while
 - c. for Loop
- F. Functions
 - 1. Syntax
 - 2. Return Values
 - 3. Pass by Value vs. Pass by Reference
 - 4. Declaration vs. Definition

III. Pointers

A. Arrays

1. Fixed size, uniform data type
2. Pointer to first element address

B. Character Sequences

C. Pointers

1. Reference (&) and dereference (*) operators
2. Pointer arithmetic
3. Safe pointers
4. void pointers vs. null pointers
5. Function pointers
6. Relationship to arrays

IV. Memory Management

A. Dynamic Memory

1. Allocation with new
2. Deallocation with delete

B. Data Structures

1. structs and pointers to structs
2. Nesting structs

C. Other Data Types

1. Aliases
2. Unions
3. Enumerations

V. Classes and Methods

A. C++ Classes

1. Access specifiers
2. Scope operator
3. Constructors
4. Overloading (constructors and methods/functions)

B. Special Members

1. Operator overloading
2. this
3. Static members
4. Constant members
5. Template classes
6. Destructors
7. Copy constructors
 - a. Shallow copy vs. deep copy

VI. Object Oriented Programming

A. Friendship

1. Friend functions and friend classes

B. Inheritance

1. Access permissions
2. What gets inherited (and what doesn't)

C. Multiple Inheritance

D. Polymorphism

1. Concept / definition
2. Dynamic allocation and polymorphism

- E. Virtual Members
- F. Abstract Base Classes

VII. File Input/Output (I/O)

- A. ofstream, ifstream, fstream
- B. Modes of opening a file
- C. Text files vs. binary
- D. get and put positions
- E. Buffers and writing file to disk

VIII. Multi-File Programs

- A. Multi-File Programs
 - 1. Header files
 - 2. Command line compilation
 - 3. Compiling and linking
- B. makefiles
 - 1. Creating and using variables
 - 2. Rules and dependencies
 - 3. Running a makefile
 - 4. Running a single rule in a makefile