COURSE NUMBER: CSCI225 **CREDITS: 3**

COURSE TITLE: Data Structures and Programming **PREREQUISITES:** CSCI 125

Weekly Hours: 4	**Lecture: 2.5**	**Lab: 1.5**	**Total Hours: 4**	**Total Weeks: 13**

COURSE DESCRIPTION: This course will delve into data structures and how they assist us in solving complex problems. Topics include: abstract data types (abstraction, encapsulation, information hiding), data structures (arrays, lists, stacks, queues, trees, heaps, hash tables, and graphs), searching and sorting algorithms, correctness and efficiency, and object-oriented programming. Programming will be done using Java as programming language.

TEXTBOOK: Tony Gaddis, Godfrey Muganda: Starting Out with Java: From Control Structures through Data Structures, 3rd Edition. Pearson. 2016. ISBN 0134038177 • 9780134038179.

LEARNING OUTCOMES:

Upon successful completion of this course, students should be able to:

- Explain the fundamentals of data structures.
- Understand and apply the concept of abstract data types (ADTs).
- Use elementary data structures from the Java API such as arrays and lists, to solve problems.
- Demonstrate skills in tracing, analyzing, and designing recursive algorithms and recursive methods.
- Understand the purpose of algorithm analysis and be able to apply this to determine the running-time of simple non-recursive algorithms. Define and use big-O notation.
- Understand and implement various sorting algorithms and analyse the running-time required to determine their efficiencies.
- Describe, analyse and implement linear search and binary search algorithms.
- Describe how to use generic classes and methods in Java.
- Demonstrate how and when to use a linked list to store elements. Write code for basic operations such as add, insert, remove, and traverse.
- Understand the doubly linked list data structure.
- Describe what a stack and what a queue is. Understand how to write array-based and linked list-based stack and queue classes to solve small but realistic problems.
- Describe in-depth applications of other data types such as trees (binary, binary search and AVL) and heap.
- Describe and implement queues and priority queues for solving programming problems.
- Understand the characteristics and optimal behaviour of hash tables for access and retrieval.
- Explain the basics of graphs and their applications.

COURSE CONTENT:

Week	Topic
Week 1	Introduction to Data Structures, Abstract Data Types, Array and ArrayList class
Week 2	Introduction to Stacks, their Applications and Implementation. Generics
Week 3	Introduction to Queues, their Applications and Implementation
Week 4	Introduction to Algorithms, Algorithm Analysis and O notation. Searching Algorithms
Week 5	Sorting Algorithms
Week 6	Midterm Exam. Advanced Sorting Algorithms
Week 7	Linked Lists and their Implementations
Week 8	Binary Trees
Week 9	AVL trees
Week 10	Hashing
Week 11	Priority queues, their Applications and Implementation
Week 12	Heapsort
Week 13	Introduction to Graphs

EVALUATION:

Participation (Attendance and Lab work)	15%
Assignments	20%
Midterm Exam	25%
Final Exam	40%
Total	100%

PLAGARISM, ACADEMIC HONESTY, AND EXAMS:

Students must hand in their own work in class assignments and projects. Passing on someone else's ideas, writings, or programs as one's own solutions in assignments, without citing the original source, is considered as plagiarism and as academic dishonesty. Those submissions will be marked as 0%.

Submissions of students who shared their work with other students (unless on team projects) will also be marked as 0%.

If a student misses an exam, a mark of 0% will be assigned unless there are extenuating circumstances. In such cases, the proportion of the grade assigned to the missed exam will be added to the proportion assigned to the final exam. The final exam will be held during the exam week. No consideration, except medical conditions, will be given to any student wishing to write the exam at any other time than the assigned one.

It is a student's responsibility to know and follow the college's policies regarding cheating on exams as outlined in the student handbook.

Unauthorized electronic devices such as calculators, smart phones, or smart watches are not allowed during exams according to the college's policy. A student who has an unauthorized electronic device on their person or around their desk is considered guilty of cheating and an "F" grade might be given for entire course.