## **COSC 248: Algorithms and Complexity (3 credits)**

The course covers topics on technical analysis and runtime algorithms, asymptotic notation, some data structures such as graphs, sorting and searching algorithms and algorithm design techniques. The course covers the application of standard algorithmic approaches, including greedy, divide and conquer, and dynamic programming. Students will also learn about basic graph algorithms and NP-completeness. *(Prerequisite: MATH 203, COSC 125)*

**Course Learning Outcomes:**

By the end of the course, students will be able to:

A1. Demonstrate advanced knowledge and understanding of classic and some emerging algorithms, complexity and methods.

A2. Appropriately adapt a variety of algorithmic problem-solving strategies to real world problems.

B1. Evaluate the time and space complexity of algorithms, using asymptotic notation.

B2. Implement a range of advanced efficient algorithms using a programming language.

**Course Learning Materials:**

* Jon Kleinberg and Éva Tardos (2006). Algorithm Design. Pearson/Addison-Wesley
* Grokking Algorithms: An Illustrated Guide for Programmers and Other Curious People. 1st Edition by Aditya Bhargava
* Computer Science Distilled: Learn the Art of Solving Computational Problems. by Wladston Ferreira Filho

**Course Content:**

1. Algorithm analysis and asymptotic notation
2. Graphs & Graph Algorithms
3. Greedy Algorithms
4. Divide & Conquer Algorithms
5. Dynamic Programming Algorithms
6. Network Flow Algorithms
7. NP and computational intractability
8. Reductions