CADT IDT វិទ្យាស្ថានបច្ចេកវិទ្យាឌីជីថល Institute of Digital Technology

Course Syllabus

Next-Gen Engagement Program – Batch II

Course Tittle : Advanced Algorithm
Timeline : July 28 – August 31 2025 (5weeks)
In-class hours : 3h/week (15h in total)
Prepared by : NEXT-GEN Engagement Team

1. Course description

Welcome to Advanced Algorithm!

In this course, you will gain comprehensive understanding of Introduction to C++ programming, algorithms, their design principles. Through a hands-on and engaging learning experience, you will explore key algorithms strategies.

Throughout this course, you will be able to analyze algorithm efficiency, implement various algorithm in C++, foster a collaborative learning environment, and get ready to explore and transform your ideas into real-world problem.

2. Course Learning Outcomes

By the end of the course, you should gain the following outcomes:

Knowledge
- Understand C++ basic syntax and structure - Explain key concepts of algorithm design and complexity analysis - Identify and describe different algorithmic strategies
Skills
- Implement basic C++ programming - Implement algorithms in C++ to solve specific problems - Analyze and compare algorithm efficiency using Big O notation - Apply appropriate algorithm techniques based on problem requirements
Attitudes
- Foster teamwork and peer collaboration in practice activities - Demonstrate problem-solving mindset with logical thinking

3. Course sessions

What we will go through each week

W1: Getting Started & Loops		
S1	Learning	- Understand the basic syntax of C++ - Learn about data types (int, float, char, bool, …) - Use input/output (cin, cout) - Apply basic math and comparison operators - Learn simple decision making using if and else - Understand how for and while loops work
S2	Practice	- Write your first C++ program and play around with another data types and if, else condition, for and while loops
W2: Functions & Arrays and Strings		
S1	Learning	- Learn to define and call simple functions - Understand return values and parameters - Learn what arrays are and how to use them - Work with basic string handling using std::string
S2	Practice	- Write function return or non-return - Store and print a list using arrays
W3: File I/O and Pointers		
S1	Learning	- Learn to open, read, write, and close text file - Understand safe file handling techniques
S2	Practice	- Create a program to save and load from a file
W4: Introduction to Class		
S1	Learning	- Understand what a class is and why it's useful - Learn to define a simple class with attributes and methods - Understand object creation and member access -
S2	Practice	- Create a class with properties - Create an object of the class and print its values
W5: Data Structures & Big O Notation		
S1	Learning	- Learn how Dynamic arrays, Linked lists, Stacks, and Queues work - Compare each structure in terms of memory, performance, and use cases
S2	Practice	- Implement a dynamic array, linked list, stack, and queue - Perform insertion and deletion operations in each

		structure - Create real-world mini problems to apply each structure

Note: The weekly content are flexible and may be adjusted to better suit students' performance and learning pace .

4. Resources

Course books

- Problem Solving in Data Structures & Algorithms Using C++ By Hemant_Jain_2016
- Learning Algorithms Through Programming and Puzzle Solving
- Grokking-Algorithms(edu.anarcho-copy.org/Algorithm/grokking-algorithms-illustrated-programmers-curious.pdf) By Aditya Bhargava – 2016
- Cracking-the-Coding-Interview

Data Structure & Algorithm Resources

- Data Structures and Algorithms Tutorial

C++ Resources

- cplusplus.com/doc/tutorial/
- cplusplus.com/reference/
- Google C++ Style Guide

Tools

- Online C++ debugger and visualizer (Online C++ Compiler, Visual Debugger, and AI Tutor - Learn C++ programming by visualizing code)
- Online C++ IDE (C++ Online Compiler)