Object Oriented Programming

(UOMU0202031)

البرمجة الكيانية
2026-2025

Week 1
Module Introduction

by
Dr Murtada Dohan
murtada.dohan@uomus.edu.iq

Agenda

- Module Aim

- Learning Outcomes

- Module outline

- Assessments

- Development Environment

- Resources

Who, Where and When?

- **Dr Murtada Dohan**, Researcher and Lecturer in College of Engineering.
  - murtdada.dohan@uomus.edu.iq use [Programming Essentials] prefix in email subject
- Education:
  - BSc Software Engineering (University of Northampton, UK)
  - MSc Software Engineering (University of Northampton, UK)
  - PhD in AI & VR (University of Northampton, UK)

- **Teaching Expertise**
  - Artificial Intelligence, Computer Vision, Data Structures..

- **Research**
  - Virtual Reality, Human Attention, Machine Learning

- **Supervision**
  - Accepting dissertation students

Module Aim

- Understand and apply object-oriented programming principles.
- Design and implement object-oriented solutions to programming problems.
- Utilize C++ libraries and frameworks for application development.
- Implement data abstraction and encapsulation for secure and efficient code.
- Plan and execute testing strategies for reliable programs.
- Debug and optimize program performance for efficient execution

Learning Outcomes

- Demonstrate a clear understanding of object-oriented programming principles, including inheritance, polymorphism, and encapsulation.

- Design and implement classes and objects to represent real-world entities, applying appropriate inheritance and encapsulation.

- Implement data abstraction and encapsulation techniques to ensure secure and efficient code.

- Identify and debug program errors using appropriate tools and techniques, enhancing program robustness.

- Identify and debug program errors using appropriate tools and techniques, enhancing program robustness.

- Evaluate and optimize program performance through code analysis and profiling, improving execution efficiency.

- Collaborate effectively with peers to develop object-oriented solutions to complex programming challenges.

- Apply exception handling techniques to handle errors and ensure program stability.

- Demonstrate proficiency in utilizing debugging tools to identify and fix program errors.

Module outline

week semesters	Topic
Week 1	Introduction to Object-Oriented Programming
Week 2	Classes, Objects, and Relationships
Week 3	Inheritance and Polymorphism principles
Week 4	Encapsulation and Data Abstraction
Week 5	Problem Analysis and Requirements Gathering
Week 6	Object-Oriented Design Principles and Patterns
Week 7	**Mid-term Exam**
Week 8	C++ Language Essentials and Advanced Topics
Week 9	Implementing Classes and Objects in C++
Week 10	Implementing Inheritance and Polymorphism in C++
Week 11	Handling Exceptions in C++
Week 12	Utilizing C++ Libraries and Frameworks
Week 13	Testing Methodologies and Strategies in C++
Week 14	Debugging Techniques and Tools in C++
Week 15	Optimization and Performance Analysis in C++
Week 16	**Preparatory week before the final Exam**

Assessments

- **Formative 40%**
  - Quizzes x 2
  - Project x 1
  - Homework (Assignments) x 2
  - Reports x 1
- **Mid**
  - 10%
- **Final**
  - 50%

Development Environment

- https://visualstudio.microsoft.com/downloads/
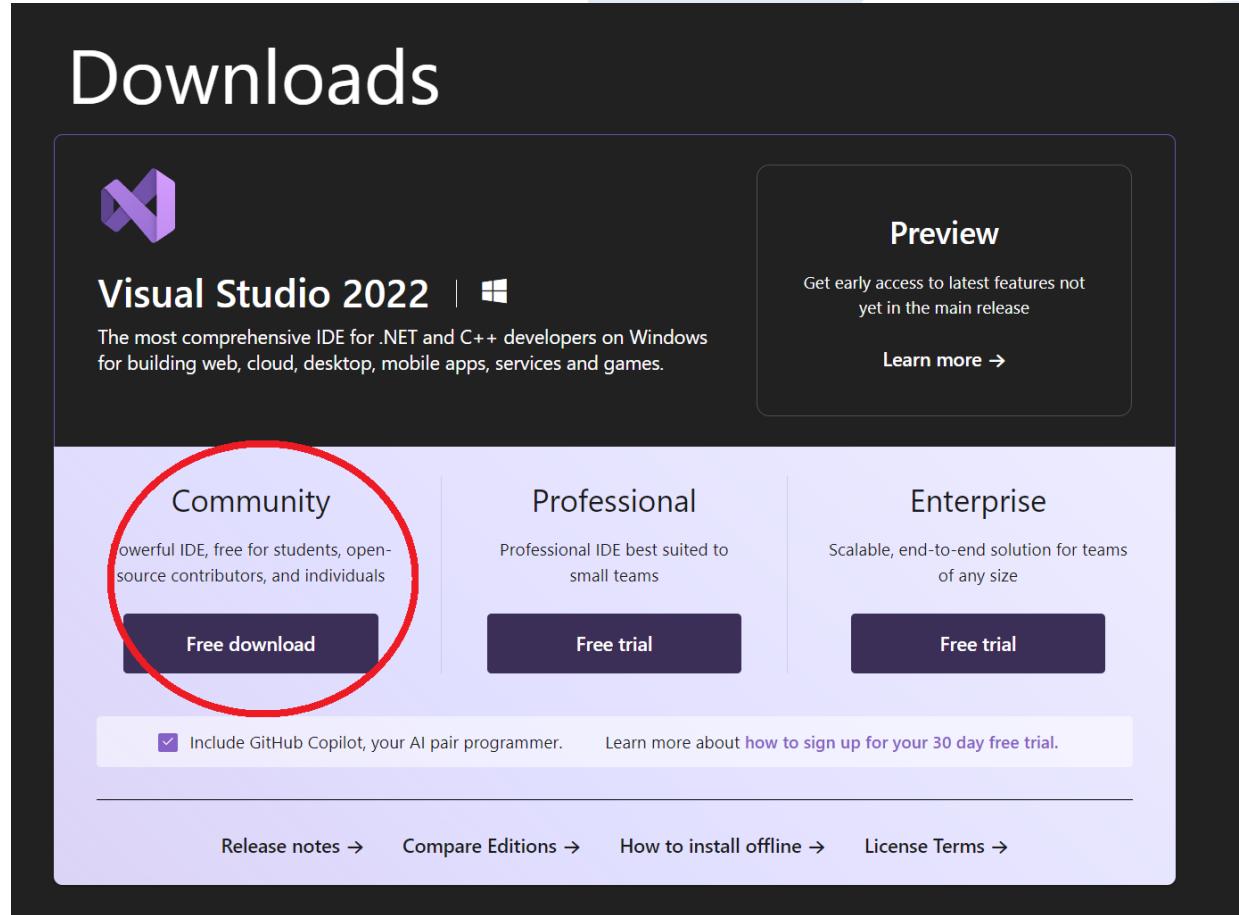
IDE Options (Free Download)

- Visual Studio (Community Version) – Windows only

- Code::Blocks (Windows only)

- Xcode (Mac only)

- CodeLite (Windows/Linux/Mac)

- Eclipse

- Online C++ Shell (cpp.sh,...)
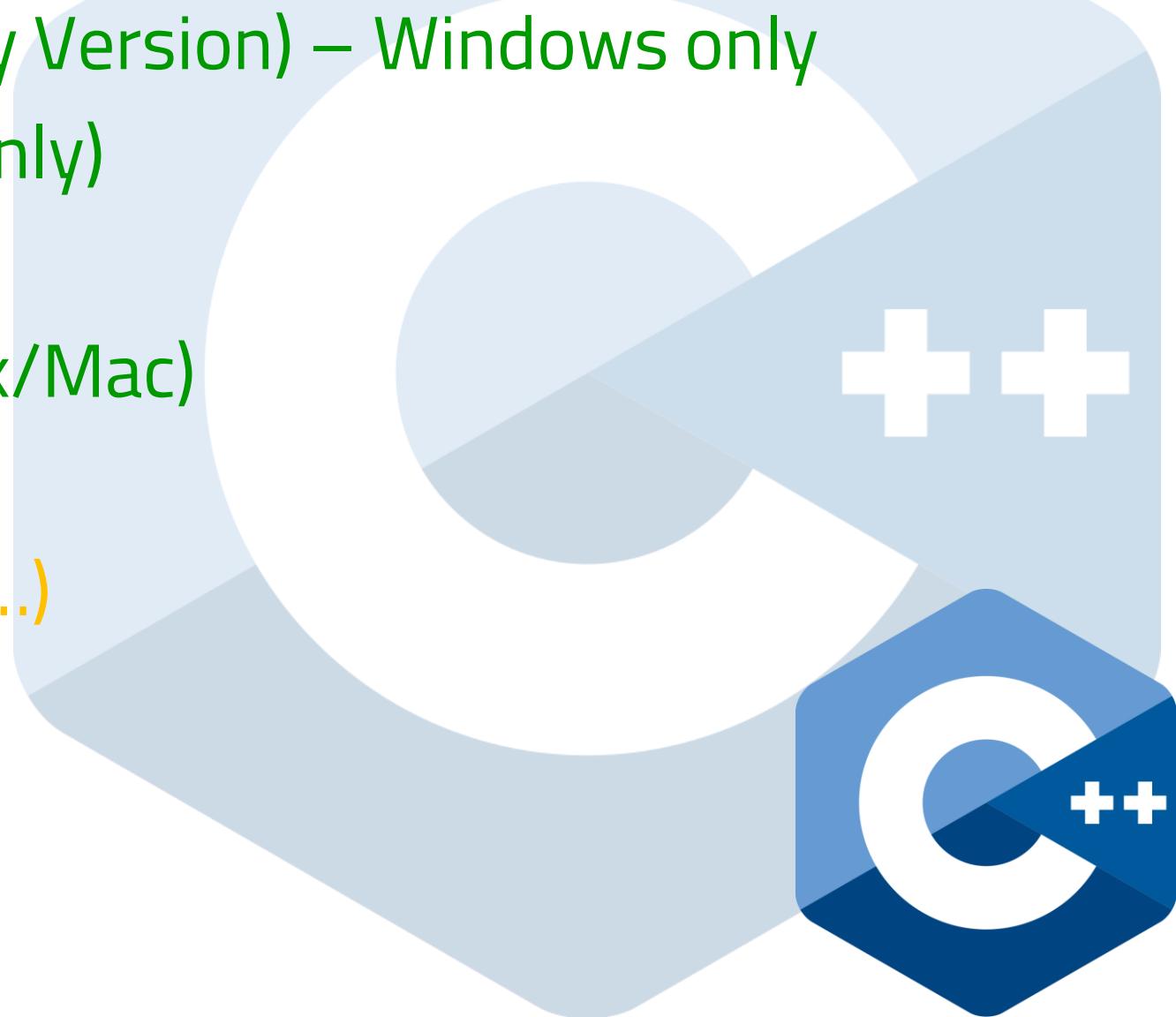
Resources

- Books:
  - C++ How to Program, 6th Edition 2007 By P. J. Deitel - Deitel & Associates, Inc., H. M. Deitel - Deitel & Associates, Inc.
  - Starting Out with Programming Logic and Design (What's New in Computer Science), 5th Edition 2018 By Tony Gaddis
- Websites:
  - https://www.w3schools.com/cpp
  - https://www.geeksforgeeks.org/c-plus-plus
  - https://en.cppreference.com/
  - https://www.w3schools.com/cpp/cpp_oop.asp
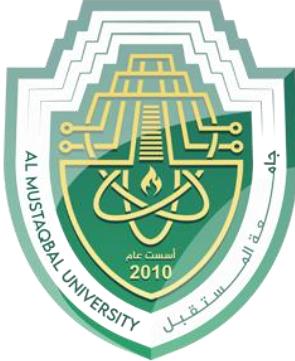
Let's try C++

Install Visual Studio (VS) and familiarise yourself with its interface.