

PROGRAMMING WITH C++

1. General Information

Course name: Programming with C++

Course code: INT1339_CLC

Number of credits: 3

2. Objectives

Knowledge:

The objective of this course is to provide learners with programming skills in the C ++ language. Learners can use C++ programming language as an important tool to solve specific problems in computer science.

Skills:

On successful completion of this course a learner will:

- make use of tools such as Dev-C++ in implementing C++ programs.
- apply knowledge and skills of C++ language for implementing C++ programs, simple object-oriented programs.
- apply knowledge and skills of C++ library STL for implementing programs more effectively.

Attitude:

Learners are required to attend the classes and complete assignments/projects.

3. Abstracts

C++ programming language is one of the most popular languages used by programmers. In this course, learners are fully equipped with basic programming skills, STL library, object-oriented programming and advanced programming in the C++ language. Proficient in programming C++, learners can use it in solving real-world problems.

4. Teaching and learning methods

- Lectures: 30h
- Exercises: 08h
- Projects: 00h
- Lab: 06h
- Individual reading: 01h

5. Prerequisites

6. Learning outcomes

After studying this course, the studier could:

[LO1]: understand knowledge and apply the requirement skills in structured programming methods of C ++ language

[LO2]: understand knowledge and apply the requirement skills in STL library

[LO3]: understand knowledge and apply the requirement skills in object-oriented programming methods of C ++ language, input/output file and exception handling

7. Assignment criteria /Learning outcomes matrix

Learning outcomes	Assignment criteria
[LO1]: understand knowledge and apply the requirement skills in structured programming methods of C ++ language	Chapter 1
[LO2]: understand knowledge and apply the requirement skills in STL library	Chapter 2
[LO3]: understand knowledge and apply the requirement skills in object-oriented programming methods of C ++ language, input/output file and exception handling	Chapter 3 Chapter 4

8. Outlines

Chapter 1. C++ Basics

- 1.1. Introduction
- 1.2. C++ environment setup
- 1.3. C++ basic syntax
- 1.4. Debug and execute a C++ program
- 1.5. C++ variables and simple data types
- 1.6. C++ operators
- 1.7. C++ control statements
- 1.8. C++ data types
 - 1.8.1. arrays
 - 1.8.2. strings
 - 1.8.3. pointers.
 - 1.8.4. references
 - 1.8.5. structures
- 1.9. C++ functions
- 1.10. Summarization and Exercise

Chapter 2. STL library

- 2.1. Container and iterator
 - 2.1.1. Sequence containers
 - 2.1.2. Iterator
 - 2.1.3. Other containers
- 2.2. Algorithms
 - 2.2.1. Non-modifying sequence operations
 - 2.2.2. Modifying sequence operations

- 2.2.3. Sorting and Searching
- 2.2.4. Others
- 2.3. Other functions
- 2.4. Summarization and Exercise
- Chapter 3. C++ object oriented**
- 3.1. C++ classes and objects
 - 3.1.1. C++ class definitions
 - 3.1.2. Define C++ objects
 - 3.1.3. C++ class member functions
 - 3.1.4. C++ class access modifiers
 - 3.1.5. C++ class constructor and destructor
 - 3.1.6. C++ friend functions
 - 3.1.7. C++ inline functions
 - 3.1.8. C++ this pointer
 - 3.1.9. Pointer to C++ classes
 - 3.1.10. Static members of a C++ class
- 3.2. C++ Inheritance
 - 3.2.1. Base and derived classes
 - 3.2.2. Access control and inheritance
 - 3.2.3. Type of inheritance
 - 3.2.4. Multiple inheritance
- 3.3. C++ Overloading
 - 3.3.1. Function overloading in C++
 - 3.3.2. Operators overloading in C++
 - 3.3.3. Overloadable/non-overloadable operators
- 3.4. Polymorphism in C++
- 3.5. Data abstraction in C++
- 3.6. Data encapsulation in C++
- 3.7. Interfaces in C++ (Abstract classes)
- 3.8. Summarization and Exercise

Chapter 4. Input/output with File and exception handling

- 4.1. C++ Files and streams
 - 4.1.1. Ofstream
 - 4.1.2. Ifstream
 - 4.1.3. Fstream
- 4.2. C++ Exception handling
 - 4.2.1. Throwing exceptions
 - 4.2.2. Catching exceptions
 - 4.2.3. C++ standard exceptions
- 4.3. Summarization and Exercise

9. Required Textbooks

- [1]. Bjarne Stroustrup, *The C++ Programming Language*, 4th Edition, 2014.
- [2]. Nicolai M Josuttis, *The C++ Standard Library: A Tutorial and Reference*, 2017.

10. Suggested Textbooks

- [3]. Deitel, Paul J, *C++ How to program*. Pearson, 10th edition, 2017.

11. Schedules

Main contents	Duration	Specific contents
Chapter 1. C++ Basics	6h lecture 2h exercise	1.1. Introduction 1.2. C++ environment setup 1.3. C++ basic syntax 1.4. Debug and execute a C++ program 1.5. C++ variables and simple data types 1.6. C++ operators 1.7. C++ control statements 1.8. C++ data types 1.9. C++ functions 1.10. Summarization and Exercise
Chapter 2. STL library	6h lecture 2h exercise 2h lab	2.1. Container and iterator 2.2. Algorithms 2.3. Other functions 2.4. Summarization and Exercise
Chapter 3. C++ object oriented	12h lecture 2h exercise 2h lab	3.1. C++ classes and objects 3.2. C++ Inheritance 3.3. C++ Overloading 3.4. Polymorphism in C++ 3.5. Data abstraction in C++ 3.6. Data encapsulation in C++ 3.7. Interfaces in C++ (Abstract classes) 3.8. Summarization and Exercise
Chapter 4. Input/output with File and exception handling	4h lecture 2h exercise 2h lab	4.1. C++ Files and streams 4.2. C++ Exception handling 4.3. Summarization and Exercise
Summary	2h lecture	

12. Grading Policy

Attendance:	10%
Exercises:	20%
Mid-term exams:	20%
Final examination (lab):	50%