distancelearningcentre

C++ (.NET) COURSE

C++ (.NET)

The **C++ (.NET) Course** covers basic programming concepts and teaches students how to build a program using the C++ .NET programming language.

Students will learn the differences between low-level and high-level languages. They will also learn how to use C++ to program variables, constants, control structures, value-returning and void functions, selection structures, loops, and arrays. Students will learn how to build, execute, and debug a C++ program, as well as how to implement sequential access files and access data from a database.

This course also covers object-oriented programming concepts, such as classes and objects. The manual is designed for quick scanning in the classroom and filled with interactive exercises that help ensure student success.

Price:
£125.00

Instalment Options:
You can spread the payments for this course over 4 monthly payments. 1 initial payment of £50.00, followed by 3 monthly payments of £25.00.

Course Format:
Course Book & CD ROM

Assessment:
No Assessment

Approximate Study Time:
80 Hours of Self Study

Course Objectives:

On completion of this course students will be able to:

- Identify the types of hardware and software and the types of programming languages.
- Distinguish between the sequence, selection, and repetition structures and write algorithms to implement control structures.
- Identify problem-solving techniques and analyze, plan, and desk-check an algorithm.
- Distinguish between a variable, a named constant, and a literal constant, use variables to store data in memory and interact with a user by using input and output methods.
- Apply programming skills to resolve a case study and build an application in Visual Studio .NET.
- Describe value returning functions and implement value returning functions.
- Define void functions and pass variables by reference to a function and use void functions in resolving a case study.
- Describe the features of selection structure and implement selection structure in a case study.

The course is split into the following Units:

- **Unit 1: Computers & Programming Languages**

The following topics are covered by this unit: Components of a PC system; and History of programming languages.

- **Unit 2: Control Structures**

The following topics are covered by this unit: Introducing control structures; and Applying control structures.

- **Unit 3: Problem Solving**

The following topics are covered by this unit: Problem-solving techniques; and Building an algorithm.

- **Unit 4: Programming Basics**

The following topics are covered by this unit: Variables and constants; Working with variables; and Input and output methods.

- **Unit 5: Building an Application**

The following topics are covered by this unit: Program construction; and Creating and managing a project.

- **Unit 6: Value-Returning Functions**

The following topics are covered by this unit: Functions; and Implementing value-returning functions.

- **Unit 7: Void Functions**

The following topics are covered by this unit: Introducing void functions; and Implementing void functions.

- **Unit 8: Selection Structures**

The following topics are covered by this unit: Introducing selection structures; and Implementing selection structures.

- **Unit 9: Nested Selection Structures**

The following topics are covered by this unit: Introducing nested selection structures; Multiple-path selection structures; and Implementing nested selection structures.

- **Unit 10: Pre-Test Loops**

The following topics are covered by this unit: Introducing pretest loops; and Applying pretest loops**.**

- **Unit 11: Post-Test Loops**

The following topics are covered by this unit: Introducing posttest loops; and Applying posttest loops**.**

- **Unit 12: Object-Orientated Programming**

The following topics are covered by this unit: Introducing object-oriented programming; and Implementing object-oriented programming**.**

- **Unit 13: Sequential Access Files**

The following topics are covered by this unit: Introducing sequential access files; Writing and reading sequential access files; and Implementing a sequential access file.

- **Unit 14: Arrays**

The following topics are covered by this unit: Introducing arrays; and Implementing arrays**.**

- **Unit 15: Advanced Arrays**

The following topics are covered by this unit: Understanding advanced arrays; and Implementing String arrays**.**

- **Unit 16: Accessing Data from a Database**

The following topics are covered by this unit: Introducing data access; and Implementing database access**.**

Pre-Requirements:

Students should have a good working knowledge of other Programming Languages.

Course Duration & Support:

Students may register at any time. The courses are designed as self-study courses but if you have any problems you can email our email support. As the course is self study you can complete in as little or as long a time as you prefer, and we do not impose a cut-off date for study.

Assessment:

The course has non-assessed assignments to complete.

Qualification:

There is no certification on completion of the course.