Course : Learn to program (with Visual Basic)

Development novices learn the basics of algorithmics
Practical course - 3d - 21h00 - Ref. INP
Price : 1940 CHF E.T.

This course introduces you to the basics of programming and algorithms. You'll work through the various key stages in building a computer program, using the Visual Basic language. You'll learn about lexicon and syntax, tools, code organization and testing.

Teaching objectives

At the end of the training, the participant will be able to:

- Structuring programs according to an algorithm
- Master the lexicon and syntax of a language to write a program
- Compiling and running a program
- Debugging and testing a program
- Accessing a database
- Understand the main principles of Object-Oriented Programming

Intended audience
Anyone who needs to learn programming.

Prerequisites
No special knowledge required.

Practical details

Hands-on work
This algorithms course contains over 60% of practical work using the Visual Basic language. However, this is not a VB course.

Teaching methods
Active pedagogy based on examples, demonstrations, experience sharing, case studies and assessment of learning throughout the course.

Course schedule

PARTICIPANTS
Anyone who needs to learn programming.

PREREQUISITES
No special knowledge required.

TRAINER QUALIFICATIONS
The experts leading the training are specialists in the covered subjects. They have been approved by our instructional teams for both their professional knowledge and their teaching ability, for each course they teach. They have at least five to ten years of experience in their field and hold (or have held) decision-making positions in companies.

ASSESSMENT TERMS
The trainer evaluates each participant's academic progress throughout the training using multiple choice, scenarios, hands-on work and more.
Participants also complete a placement test before and after the course to measure the skills they've developed.

TEACHING AIDS AND TECHNICAL RESOURCES
• The main teaching aids and instructional methods used in the training are audiovisual aids, documentation and course material, hands-on application exercises and corrected exercises for practical training courses, case studies and coverage of real cases for training seminars.
• At the end of each course or seminar, ORSYS provides participants with a course evaluation questionnaire that is analysed by our instructional teams.
• A check-in sheet for each half-day of attendance is provided at the end of the training, along with a course completion certificate if the trainee attended the entire session.

1 The foundations of programming

- What is a program? What is a language? The different paradigms. Which language for which application?
- Compilers. Executables.
- The responsibilities of a programmer.
- What is an algorithm?
- The needs met by an algorithm.
- The concept of pseudo-language.

Hands-on work
Presentation of different languages (Java, C#, Visual Basic, C, C++). Write your first algorithm in a pseudo-language.

2 Genesis of a first program

- Write a simple program: syntax and instructions.
- Compile and run the program.
- What is a bookshop? Its role, its use.

Hands-on work
Discover the development and execution environment. Write, compile and run a first program.

3 Programming rules

- Naming convention.
- Syntax convention.
- Using comments. Why comment on developments?
- Improve program readability: code indentation, code breakdown, etc.

4 Variables

- What is a variable?
- Why type a variable?
- Primitive types: integers, strings, real numbers, others.
- Declaration, definition and initialization of a variable.
- Constants.
- Input, display, assignment, type conversion.
- Organize data in tabular form.
- Advanced types: record, matrix, tree.

Hands-on work
Write several simple programs to manipulate variables.

5 Operators and expressions

- The different operators (multiplicative, additive, comparison, equality, logic, assignment).
- Combination of operators.
- Boolean expression.

Hands-on work
Operators and Boolean expressions.

(6) Control structures

- Alternative selections (if, if-then-no, case selection).
- Insctructions blocks (notion of beginning... end).
- Iterative loops (while-repeat, repeat-until, for-from-to).
- Instruction nesting.
- Comments.

Hands-on work
Use control structures to implement an algorithm.

(7) Procedures and functions

- Definitions: procedure, function.
- Why are they essential to programming (reusability, readability, etc.)?
- Parameter passing.
- The return code of a function.
- Awareness of the limits of variable value passing.
- Notion of passage by address.
- Call functions.

Hands-on work
Debugging example programs.

(8) Introduction to object-oriented programming

- Object programming concepts: class, attribute, method, argument.
- Object modeling based on functional requirements.
- Introduction to best practices in program design and organization.

Hands-on work
Illustration of object concepts.

(9) Access to databases

- Data organization and storage.
- Basic processing (connection, queries, data retrieval).
- Client application and data server.
- Data display and manipulation in the client application.

Hands-on work
Create a form to search for information in a database.

(10) Program maintenance, debugging and testing

- Read and interpret error messages.
- Use a debugger: execute a program step by step, breakpoints, inspect variables during execution.
- Plan unit tests.

Hands-on work
Use a debugger to control program execution.

Dates and locations

REMOTE CLASS
2026 : 17 June, 9 Dec.