

Course : Algorithms, introduction

Practical course - 4d - 28h00 - Ref. AGW

Price : 2470 CHF E.T.

 3,8 / 5

This course gives you the basics of algorithmics, essential for writing clean, efficient programs.

Teaching objectives

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

- ✓ Knowledge of major sorting algorithms
- ✓ Knowledge of the main data structures essential to algorithmics
- ✓ Designing algorithms for graphs and trees

Intended audience

Anyone who needs to learn programming.

Prerequisites

No special knowledge required.

Practical details

Hands-on work

Training alternating theory and practice.

Course schedule

1 Elementary actions, variables and types

- Elementary actions, variables, elementary types.
- Loops, procedures and functions.
- Inheritance and exceptions. How OOP simplifies writing and reading programs/algorithms.
- Recursivity.

Hands-on work

Conception d'algorithmes avec des boucles imbriquées. Décomposition de programmes en procédures et fonctions. Conception d'algorithmes récursifs.

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.

2 Algorithms on arrays

- Syntax conventions.
- Traverse a 1-dimensional array. Calculate sum and average.
- Find a value in a table of dimension 1. Maximum value, minimum value.
- Traversing a 2-dimensional array.
- Find a value in a 2-dimensional array.

Hands-on work

Calculate the transpose of an $n \times n$ matrix.

TERMS AND DEADLINES

Registration must be completed 24 hours before the start of the training.

ACCESSIBILITY FOR PEOPLE WITH DISABILITIES

Do you need special accessibility accommodations? Contact Mrs. Fosse, Disability Manager, at psh-accueil@orsys.fr to review your request and its feasibility.

3 Sorting algorithms

- Bubble sorting.
- Sorting by selection.
- Insertion sorting.
- The heap sort.
- Merge sorting.
- The quicksort.

Hands-on work

Comparison of the maximum number of swaps between two algorithms.

4 The main data structures

- Queues; FIFO structure.
- Example of an algorithm using a FIFO structure.
- Stacks; LIFO structure.
- Example of an algorithm using a stack structure.

5 Algorithms on graphs

- Graph representation.
- Graph traversal in width.
- Search for a minimal spanning tree.
- Algorithm for finding the shortest paths between all pairs of points.

Hands-on work

Writing a minimum spanning tree search algorithm.

6 Tree traversal algorithms

- Graph representation.
- Width first. Sum calculation.
- First, an in-depth tour.
- Abandon branch exploration (back-tracking).

Hands-on work

Write an algorithm that requires deep exploration of a tree first.

Dates and locations

REMOTE CLASS

2026: 17 Mar., 9 June, 20 Oct.