

C++ Language – Object Oriented Programming

Course code: CPP_OOP

Who is the course for The course is intended for programmers, testers and project leaders who want to learn about designing and implementing objects in C++. What we teach you Definition of an object in C++ Object attributes and methods Access rights to attributes and methods Constructors, implicit constructors, copy constructors Destructors Inheritance Virtual functions, destructors New and Delete operators Static attributes and methods Required skills Skills corresponding to the C and C++ programming languages course (MSCPP1) Teaching methods Expert instruction with practical examples, computer practice Course Outline Introduction to Object Oriented Programming Objects and Classes Encapsulation of objects Class Inheritance Using polymorphism Object oriented programming Class keyword Defining attributes Naming Conventions Defining scope Defining Methods Defining Methods overriding New and Delete operators Constructors and destructors Deep and shallow copy of object Inheritance in C++ Introducing to Inheritance in C++ Samples of objects hierarchies Protected keyword Using constructor for parent object Using methods and attributes of parent object Inheritance versus aggregation Polymorphism in C++ Virtual methods Polymorphic containers Virtual destructors Abstract classes and methods Static attributes and methods Overriding operators Explicit constructors Errors and Events

Affiliate	Duration	Course price	ITB
Praha	5	20 500 Kč	50
Brno	5	20 500 Kč	50
Bratislava	5	900 €	50

The prices are without VAT.

Course terms

Date	Duration	Course price	Type	Course language	Location
24.02.2025	5	20 500 Kč	Online	CZ/SK	GOPAS Praha online
24.02.2025	5	900 €	Online	CZ/SK	GOPAS Bratislava online
03.03.2025	5	20 500 Kč	Presence	CZ/SK	GOPAS Praha
17.03.2025	5	20 500 Kč	Presence	CZ/SK	GOPAS Brno
12.05.2025	5	900 €	Presence	CZ/SK	GOPAS Bratislava prezenčne
16.06.2025	5	20 500 Kč	Presence	CZ/SK	GOPAS Praha

The prices are without VAT.

Who is the course for

The course is intended for programmers, testers and project leaders who want to learn about designing and implementing objects in C++.

What we teach you

Definition of an object in C++
Object attributes and methods
Access rights to attributes and methods
Constructors, implicit constructors, copy constructors
Destructors
Inheritance
Virtual functions, destructors

GOPAS Praha
Kodaňská 1441/46
101 00 Praha 10
Tel.: +420 234 064 900-3
info@gopas.cz

GOPAS Brno
Nové sady 996/25
602 00 Brno
Tel.: +420 542 422 111
info@gopas.cz

GOPAS Bratislava
Dr. Vladimíra Clementisa 10
Bratislava, 821 02
Tel.: +421 248 282 701-2
info@gopas.sk

**GOPAS**[®]
Copyright © 2020 GOPAS, a.s.,
All rights reserved

C++ Language – Object Oriented Programming

New and Delete operators

Static attributes and methods

Required skills

Skills corresponding to the C and C++ programming languages course (MSCPP1)

Course Outline

Introduction to Object Oriented Programming

- Objects and Classes
- Encapsulation of objects
- Class Inheritance
- Using polymorphism

Object oriented programming

- Class keyword
- Defining attributes
- Naming Conventions
- Defining scope
- Defining Methods
- Defining Methods overriding
- New and Delete operators
- Constructors and destructors
- Deep and shallow copy of object

Inheritance in C++

- Introducing to Inheritance in C++
- Samples of objects hierarchies
- Protected keyword
- Using constructor for parent object
- Using methods and attributes of parent object
- Inheritance versus aggregation

Polymorphism in C++

- Virtual methods
- Polymorphic containers
- Virtual destructors
- Abstract classes and methods
- Static attributes and methods
- Overriding operators
- Explicit constructors
- Errors and Events

GOPAS Praha

Kodaňská 1441/46
101 00 Praha 10
Tel.: +420 234 064 900-3
info@gopas.cz

GOPAS Brno

Nové sady 996/25
602 00 Brno
Tel.: +420 542 422 111
info@gopas.cz

GOPAS Bratislava

Dr. Vladimíra Clementisa 10
Bratislava, 821 02
Tel.: +421 248 282 701-2
info@gopas.sk



Copyright © 2020 GOPAS, a.s.,
All rights reserved