Jazyk C++ – Objektově orientované programování v C++

Kód kurzu: CPP_OOP

Kurz je určen pro posluchače, kteří přecházejí z jazyka C na C++, nebo jen chtějí proniknout do problematiky návrhu a implementace objektů v jazyce C++. Na kurzu se naučíte definovat objekty, konstruktory a destruktory, používat virtuální funkce, dědičnost a agregaci, přetěžovat operátory a zpracovávat události a výjimky. Představované techniky jsou striktně platformově neutrální, použitelné v operačních systémech Windows, Unix, Linux apod.

Co Vás naučíme

- Definice objektu v C++
- Atributy a metody objektů
- Přístupová práva k atributům a metodám
- Konstruktory, implicitní konstruktory, copy-konstruktory
- Destruktory
- Dědičnost
- Virtuální funkce, destruktory
- Operátory new a delete
- Statické metody a atributy

Požadované vstupní znalosti

- Znalosti v rozsahu kurzu Programovací jazyk
- C a C++ - základní kurz (CPP_INTRO).

Osnova kurzu

C++ třída a zapouzdření

- Trochu objektově orientované teorie a názvosloví.
- Klíčové slovo „class" a „struct"
- Deklarace atributů, jmenné konvence
- Přístupová práva „private" a „public"
- Definice metod objektu
- „inline" funkce a metody
- „const" metody a „mutable" atributy
- Statické atributy a metody
- „friend" deklarace
- Implicitní konstruktor, destruktor
- Operátory „new" a „delete", alokace pole
- Význam kopírovacího konstruktoru a přiřazovacího operátoru
- „RULE OF THREE" a jeho důsledky
- Konverze pomocí konstruktorů s jedním parametrem, „explicit" konstruktory

Jednoduchá dědičnost

- Typy vazeb mezi objekty, trocha UML
- Generalizace vs. agregace
- Dědičnost a příklady objektových hierarchií
- Důsledky „IS KIND OF" vazby na kompatibilitu instancí
- Přístupové právo „protected" a zásady návrhu bázové třídy
- Přetěžování metod potomkem, zastínění bázové metody

Polymorfismus

- Co je to vlastně polymorfismus? Příklady hierarchií.
- Brzká a pozdní vazba, virtuální metody.
- Polymorfické kontejnery, virtuální destruktory

GOPAS Praha
Na Strži 2097/63
140 00 Praha 4 - Krč
Tel.: +420 226 201 390
info@gopas.cz

GOPAS Brno
Nové sady 996/25
602 00 Brno
Tel.: +420 530 513 590
info@gopas.cz

GOPAS Bratislava
Dr. Vladimíra Clementisa 10
Bratislava, 821 02
Tel.: +421 902 903 132
info@gopas.sk

Copyright © 2026 GOPAS, a.s.,
All rights reserved

- Statický vs. Dynamický typ, „dynamic_cast" a ostatní operátory pro
- přetypování
- Ryze virtuální metody a abstraktní třídy
- Pojem „interface", příklady využití

Vícenásobná dědičnost

- Výhody vícenásobné dědičnosti
- Problém s diamantem v dědičnosti
- Virtuální dědičnost, dekompozice instance do acyklického grafu
- Nejednoznačnost symbolů a inicializace instance

Výjimky

- Ošetřování chyb ve stylu jazyka C a C++
- Klíčová slova „throw", „try", a „catch"
- Návrh tříd sloužící jako výjimky, rozhraní třídy „std::exception"
- Výjimky a citlivý kód konstruktoru a destruktoru

GOPAS Praha
Na Strži 2097/63
140 00 Praha 4 - Krč
Tel.: +420 226 201 390
info@gopas.cz

GOPAS Brno
Nové sady 996/25
602 00 Brno
Tel.: +420 530 513 590
info@gopas.cz

GOPAS Bratislava
Dr. Vladimíra Clementisa 10
Bratislava, 821 02
Tel.: +421 902 903 132
info@gopas.sk

Copyright © 2026 GOPAS, a.s.,
All rights reserved