# Optimalizace výkonu, ladění a škálovatelnost aplikací

Kód kurzu: GOC2126

Kurz je určen vývojářům na platformě .NET (Core), kteří chtějí proniknout do detailů .NET a C# aplikací a naučit se psát maximálně rychlé aplikace, používat správné konstrukty nebo optimalizovat spotřebu paměti. Zároveň také na odhalování a diagnostiku těchto problémů (post-mortem i aktuálních) s použitím dostupných nástrojů. Důraz je kladen na praktickou aplikaci získaných znalostí. Studenti budou schopni použít nabytých znalostí okamžitě po návratu do reálného světa. Kurz je vhodný pro vývojáře nad .NET Frameworkem, .NET Core, .NET 5+, Xamarin atd.

| Pobočka | Dnů | Cena kurzu | ITB |
|---|---|---|---|
| Praha | 4 | 21 600 Kč | 40 |
| Brno | 4 | 21 600 Kč | 40 |
| Bratislava | 4 | 920 € | 40 |

Uvedené ceny jsou bez DPH.

## Termíny kurzu

| | Datum | Dnů | Cena kurzu | Typ výuky | Jazyk výuky | Lokalita |
|---|---|---|---|---|---|---|
| | 07.04.2025 | 4 | 21 600 Kč | Teleprezenční | CZ/SK | Gopas Brno Prezenční_GTT |
| | 07.04.2025 | 4 | 21 600 Kč | Teleprezenční | CZ/SK | Gopas Praha Prezenční_GTT |
| | 07.04.2025 | 4 | 920 € | Teleprezenční | CZ/SK | Gopas Bratislava Prezenční_GTT |

Uvedené ceny jsou bez DPH.

### Požadované vstupní znalosti

- Kurz předpokládá znalosti a zkušenosti s programováním v jazyce C# na úrovni kurzu GOC2124 a GOC2125

### Pro koho je kurz určen

- Kurz je určen pro zkušené vývojáře, kteří se chtějí posunout opět o něco dál a lépe porozumět tomu, jak se v .NETu pracuje s pamětí, jaké jsou skutečné možnosti debuggingu, jak se provádí kód, měří výkon nebo používá reflexe, ale i další pokročilá témata.

### Metody výuky

- Odborný výklad s praktickými ukázkami, cvičení na počítačích.

### Studijní materiály

- Tištěné prezentace probírané látky.

### Osnova kurzu

Práce s pamětí v .NETu

- Hodnotové vs. referenční typy
- Alokace a fungování GC (SOH, LOH, POH)
- Halda, zásobník, unmanaged halda
- Skryté alokace
- Stackalloc
- Span, Memory
- Unsafe
- In, ref structs, readonly
- Pedávání parametr
- Meření spotřeby paměti, memory leaks
- Profilery

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

- Finalizace
- Stringy
- WeakReference
- Pooling

Složitost algoritmu

- Big O
- Práce s kolekcemi

Debuggování

- Debuggování ve Visual Studiu a všechny možnosti debuggeru Visual Studia
- Práce se symboly
- WinDBG debugging
- ProcDump, dotnet dump
- Post mortem debugging
- Tvorba a analýza dumpu aplikačně
- Monitoring stavu/provozu aplikace (dotnet monitor)

Provádění kódu v prostředí .NET

- JIT kompilace, profile optimizations, ngen, ready to run
- Optimalizace v JITu, multicore JIT
- Tiered JIT, PGO
- Intrinsics, vektorizace, SIMD
- Fungování CPU a vliv instrukci na výkon
- Profilery

Měření výkonu kódu

- Benchmarking a jeho limitace
- Korektní vs. chybná měření
- Profilování
- BenchmarkDotNet
- Úvod do assembly

Reflection a její efektivní použití

Generování kódu

- IL Emit
- Expressions
- Source Generators

Lazy inicializace

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