BO·TT·EGA
IT minds

Program szkolenia:

JVM tuning

Informacje:

Nazwa:	**JVM tuning**
Kod:	**Java-tuning**
Kategoria:	Java i JVM
Grupa docelowa:	developerzy
Czas trwania:	3 dni
Forma:	50% wykłady / 50% warsztaty

Szkolenie pokrywa swym zakresem zarówno najbardziej typowe problemy jaki zagadnienia specyficzne z zakresu tuningu JVM i wysokowydajnych aplikacji.

Zalety szkolenia:

- Realne problemy i ich rozwiązania
- Trener z kilkunastoletnim doświadczeniem
- Ciekawe zadania praktyczne

BOTTEGA IT minds

Szczegółowy program:

1. Wprowadzenie do mechanizmów maszyny wirtualnej Java

- 1.1. Czym jest bytecode i jak jest interpretowany przez JVM
- 1.2. JMM (Java Memory Model), czyli jak działają watki i dostęp do współdzielonej pamięci oraz czym jest "lock contention"
- 1.3. Krótkie wprowadzenie do teorii "garbage collectors" (GC)
- 1.4. Generacyjny GC w maszynie JVM
- 1.5. Nowy algorytm GC w Oracle JVM czyli slow kilka o G1
- 1.6. Przestrzeń PerGem i alokacja pamięci poza stosem w JVM

2. Metryki i optymalizacja GC

- 2.1. Metody zbieranie metryk w Oracle JVM
- 2.2. Narzędzia i techniki analizy zachowania GC
  - 2.2.1. jmap, jhat, jstat, jstack
  - 2.2.2. VisualVM, GCViewer, MAT (Memory Analyzer Tool)
- 2.3. Omówienie dostępnych parametrów dla GC w Oracle JVM
- 2.4. Optymalizacja zachowania GC
- 2.5. Wpływ charakterystyki wydajności aplikacja na parametry GC
- 2.6. Jak wybrać odpowiedni GC dla naszych potrzeb

3. Metodologie optymalizacji wydajności aplikacji

- 3.1. Optymalizacja wykorzystania CPU
- 3.2. Optymalizacja wykorzystania pamięci
- 3.3. Wykrywanie wycieków pamięci
- 3.4. Wykrywania "lock contention"
- 3.5. Testowanie wydajności
- 3.6. Antywzorce w testowaniu wydajności i optymalizacji JVM

BOTTEGA IT minds

4. Tworzenie wysokowydajnych aplikacji na platformę Java

- 4.1. Najlepsze praktyki alokacji obiektów
- 4.2. Tworzenie wielowątkowych aplikacji
  - 4.2.1. Wykorzystanie java.util.concurrent
  - 4.2.2. Wprowadzenie do "lock free algorithms"
- 4.3. Efektywne operacje wejścia/wyjścia z wykorzystaniem Java NIO API
- 4.4. Techniki praca z obiektami java.lang.String
- 4.5. Optymalizacje wykorzystywane przez JVM o które bałeś się zapytać