# 10 Paralelní programování

## Asynchronní programování

* Asynchronní volání neblokující chod volajícího
* Např při stahování dat z DB, aby nezamrzla aplikace
* Responzivita aplikace
* Async a Await

## Paralelní programování

* Rozdělení práce do několika vláken
* Cílem je výkon a rychlost aplikace
* Např quicksort
* Aplikace běží ve vlákně = proces
* Můžeme vytvořit další vlákno a delegovat do něj činnost hlavní vlákno stále pracuje a není blokováno náročným výpočtem

## Thread

* Třída, která reprezentuje vlákno
* Aplikace může paralelně běžet na více vláknech či jádrech procesoru

Obsah obrázku text

Popis byl vytvořen automaticky

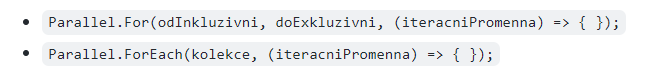
## Task

* Třída představující asynchronní operaci
* Vyšší úroveň abstrakce než thread
* Obsah obrázku text

  Popis byl vytvořen automaticky

## Parallel.For

* Paralelní průchod cyklem



## Parallel.Foreach

Obsah obrázku text

Popis byl vytvořen automaticky

Obsah obrázku text

Popis byl vytvořen automaticky

## Await async

Obsah obrázku text

Popis byl vytvořen automaticky

## Zrušení úlohy za běhu pomocí výjimky

Obsah obrázku text

Popis byl vytvořen automaticky

Obsah obrázku text

Popis byl vytvořen automaticky

## Synchronizace vláken

* Koordinovaný přístup ke sdíleným prostředkům
* Při zpracování dat vláknem by k těmto datům ostatní vlákna neměla přistupovat
  + Problém nastává, když vlákno s daty provádí neatomickou operaci a jiné vlákno začne s těmito daty také pracovat
  + (data ještě nejsou plně zpracovaná a připravená k další akci)

## Omezení přístupu do kritické sekce

* lock() uzamyká kus kódu objektem, který dostane jako vstupní parametr
* V tomto kusu kódu může být pouze jedno vlákno, ostatní čekají
* Vstupním parametrem je kód uzamčen ale samotný parametr nijak uzamčen/chráněn/modifikován není, může to být libovolný object
* Deadlock

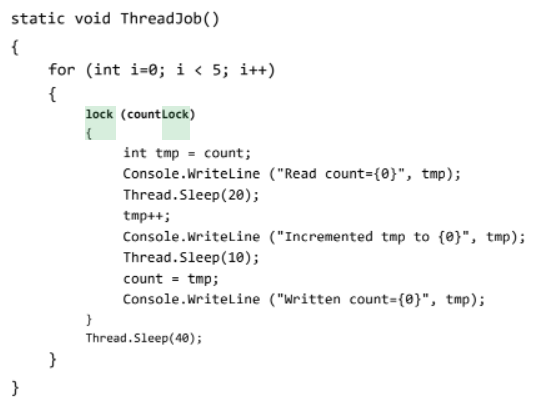
Obsah obrázku text

Popis byl vytvořen automaticky

Nebo hezčí zápis pomocí locku

Obsah obrázku text

Popis byl vytvořen automaticky



* Vzájemné uváznutí
* Přidělovat a zamykat všechny prostředky
* Metoda 1 čeká na uvolnění zdroje alokovaného metodou 2
* Metoda 2 čeká na uvolnění zdroje alokovaného metodou 2