

Warning

This page is located in archive.

1. domácí úloha: Producent – konzument

- Předloha: `pdv-01thrpool.zip` [<https://pdv.pages.fel.cvut.cz/pdv-private/hw/pdv-01thrpool.zip>]
- Do BRUTE odevzdávejte zip archiv obsahující soubor `ThreadPool.h`.

V 2. týdnu jste se v rámci přednášky a cvičení dozvěděli více o **vláknech, mutexech a podmínkových proměnných**. Nyní je na čase si nově nabyté znalosti vyzkoušet v praxi na první domácí úloze. Domácí úkol vychází z problému “**Producent – konzument**”, se kterým jste byli seznámeni rovněž na hodině.

Vaším úkolem je implementovat chybějící části kódu v souboru `ThreadPool.h` tak, abyste zajistili hladkou součinnost několika souběžně běžících vláken producentů (přidávají úlohy) a konzumentů (vykonávají úlohy), kteří přistupují ke společné frontě. Ve staženém balíčku najdete hlavní soubor `main.cpp`, který spouští celou domácí úlohu a definuje ukázkovou funkci `worker`, kterou mají počítat vlákna konzumentů. V tomto souboru nic upravovat nemusíte (ale můžete, pokud byste chtěli například vyzkoušet různé rychlosti přidávání úloh do fronty a různé počty producentů a konzumentů).

V souboru `ThreadPool.h` naleznete ukázkou realizace s jedním vláknem konzumenta, který bohužel většinou nestíhá nové úlohy ve frontě zpracovávat dostatečně rychle a úlohy se tak hromadí. Tento nedostatek lze vyřešit, tak, že se úloha bude spouštět s více vlákny konzumentů. Toho lze docílit správnou implementací konstruktoru třídy `Threadpool`, který spouští daný počet vláken konzumentů. Způsob přidávání úloh do fronty v současné implementaci není v pořádku, pokud by současným způsobem přistupovalo více vláken ke frontě současně, výsledek není deterministický. Mohlo by se stát, že vlákno přidanou úlohu “přepíše” a ta se tak ztratí.

První bod dostanete za to, pokud tyto nedostatky opravíte. Samozřejmě očekáváme, že se zpracují všechny úlohy a to právě jednou. Navíc zpracování by mělo probíhat co nejvíce nezávisle na sobě, abyste dosáhli co nejlepšího času (díky paralelnímu zrychlení).

Druhý bod dostanete, pokud vlákna konzumentů nebudou čekat na přidání nových úloh do fronty pomocí “busy-waitingu”.

Copyright © 2025 CTU in Prague | Operated by [IT Center](#) of [Faculty of Electrical Engineering](#)
| Bug reports and suggestions [Helpdesk CTU](#)