

Programowanie współbieżne

Część I: wprowadzenie i podstawy

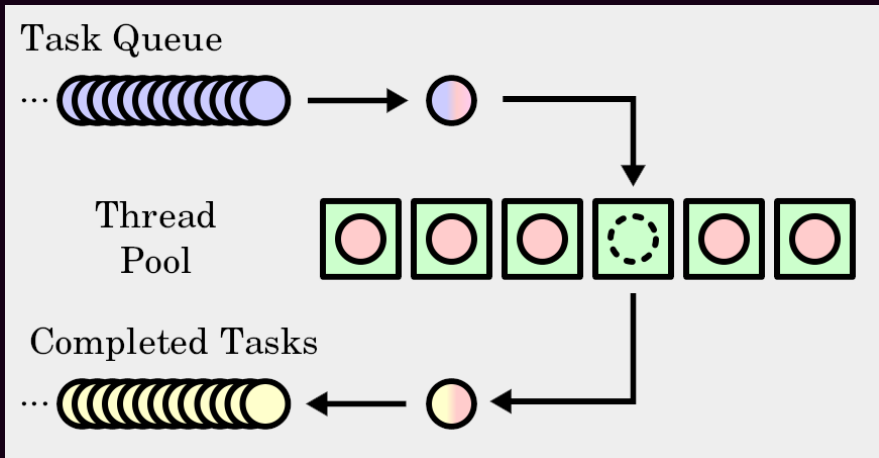
Adam Galiński dla C++ Friends

Agenda

1. Podstawy

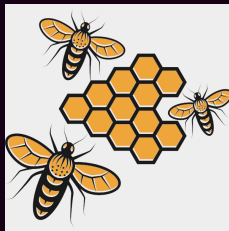
- ▶ Scheduler
- ▶ wątek (`std::thread`)
- ▶ Podstawowe problemy
- ▶ Synchronizacja
- ▶ `std::mutex`
- ▶ `std::condition_variable`
- ▶ zmienne atomowe

Scheduler



Wątki

- ▶ `std::thread(Function f, Args&&... args)`
- ▶ `join()` dołącza wątek do głównego wątku
- ▶ `detach()` dołącza* wątek do głównego wątku
- ▶ header `<thread>`. Wymagany `-std=c++11` lub nowszy.
- ▶ szybki przykład



Zadanie 0.

0. Napisz program „Hello, World!”.

- ▶ Wypisanie tekstu powinno nastąpić w dedykowanym wątku (innym niż wątek główny).
- ▶ Sprawdź różnicę w działaniu metod `join()` i `detach()`.

Zadanie 1.

1. Przeanalizuj program 01.cpp.

- ▶ Czy program jest **deterministyczny**?
- ▶ Dlaczego wewnątrz procedury `run()` korzystamy z klasy `string`?
- ▶ Jak zapewnić naprzemiennność pracy wątków roboczych?

Zadanie 2.

2. Przeanalizuj program 02.cpp.

- ▶ Czy program jest **deterministyczny**?
- ▶ Jak często wątki usiłują uzyskać dostęp do strefy krytycznej?
- ▶ Czy dostrzegasz potencjalne problemy tego kodu?

Koniec

Dzięki za uwagę!