

Deadlock (Uváznutí)

Deadlock je situace v informatice, ke které dochází při souběžném běhu více vláken. Lze ho přirovnat k situaci na křižovatce, kde do středu vjedou čtyři auta ze čtyř směrů a vzájemně si zablokují cestu – nikdo nemůže jet dál, dokud někdo jiný necouvne, ale couvnout není kam.

Čtyři podmínky pro vznik (Coffmanovy podmínky)

Aby mohl Deadlock nastat, musí být současně splněny tyto čtyři podmínky:

- Vzájemné vyloučení (Mutual Exclusion):** Prostředek (např. tiskárna nebo proměnná) může v jeden okamžik používat pouze jedno vlákno.
- Držení a čekání (Hold and Wait):** Vlákno již drží alespoň jeden prostředek a zároveň žádá o další, který momentálně drží někdo jiný.
- Nemožnost odnětí (No Preemption):** Operační systém nemůže vláknu prostředek násilím sebrat; vlákno ho musí uvolnit dobrovolně.
- Kruhové čekání (Circular Wait):** Existuje uzavřený řetězec vláken, kde každé čeká na prostředek držený dalším článkem v řetězci.

Praktický příklad: Klasický scénář

Představme si dvě vlákna (A a B) a dva zámky k datům (Zámek 1 a Zámek 2):

- **Vlákno A** získá **Zámek 1** a chce získat **Zámek 2**.
- **Vlákno B** získá **Zámek 2** a chce získat **Zámek 1**.

Obě vlákna nyní navždy stojí. Vlákno A neustoupí, dokud nedostane Zámek 2, a Vlákno B neustoupí, dokud nedostane Zámek 1.

Jak se s Deadlockem bojuje?

Existují tři hlavní strategie, jak tento problém řešit:

1. Prevence a vyhýbání se

System je navržen tak, aby jedna z Coffmanových podmínek nikdy nenastala.

- **Řazení prostředků:** Všechna vlákna musí žádat o prostředky vždy ve stejném pořadí (např. vždy nejdřív Zámek 1, pak Zámek 2). Tím se eliminuje kruhové čekání.
- **Bankéřův algoritmus:** System předem vypočítá, zda přidělení prostředku nemůže vést k nebezpečnému stavu.

2. Detekce a zotavení

System nechá Deadlock nastat, ale pravidelně kontroluje grafy závislostí. Pokud najde kruh, zasáhne:

- **Ukončení procesu:** Jedno ze zablokovaných vláken je násilně ukončeno (kill), čímž se uvolní jeho prostředky.
- **Rollback:** System vrátí proces do dřívějšího uloženého bodu (checkpointu).

3. Ignorování (Pštroší algoritmus)

Většina běžných operačních systémů (Windows, Linux) Deadlocky u uživatelských aplikací ignoruje. Předpokládá se, že k nim dochází zřídka a je levnější nechat uživatele aplikaci restartovat, než neustále monitorovat všechny prostředky.

Rozdíl: Deadlock vs. Livelock

Stav	Popis
Deadlock	Vlákna stojí a nedělají nic (jsou zablokovaná).
Livelock	Vlákna stále něco dělají (mění svůj stav), ale nikam se neposouvají (jako když se dva lidé v úzké chodbě snaží vyhnout a oba úskakují stále na stejnou stranu).

Související pojmy: Vlákno (Thread), Proces, Multitasking, Synchronizace, Mutex, Race Condition.

From:
<https://serviceit.cz/> - IT ENCYKLOPEDIE

Permanent link:
<https://serviceit.cz/doku.php?id=deadlock>

Last update: **2025/12/31 20:23**

