FAKULTA APLIKOVANÝCH VĚD
ZÁPADOČESKÉ UNIVERZITY
V PLZNI

KIV/PGS
SEMESTRÁLNÍ PRÁCE 2019/20
Práce s vlákny

Tomáš Ott
(A17B0314P)

16. dubna 2020

Obsah

1 Uživatelská příručka

Aplikaci je možné spustit z terminálu pomocí příkazu `java -jar PGS_OTT.jar <název-souboru>`. Při chodu programu nebude generován žádný výstup do terminálu. Výsledek aplikace bude zapsán do složky pod jménem `<název-souboru>` a bude obsahovat požadovanou strukturu.

Počet spuštěných vláken lze modifikovat v souboru `configuration.json`. Soubor musí obsahovat přesně dané JSON objekty jinak nebude možné nastavit aplikaci. Struktura souboru by měla vypadat následovně:

```
{
"BOSS": <počet-vláken>,
"UNDERBOSS": <počet-vláken>,
"MASTER": <počet-vláken>,
"FOREMAN": <počet-vláken>,
"SLAVE": <počet-vláken>
}
```

2 Kritické sekce

2.1 Vstupní soubor

Práce se vstupním souborem je jedna z prvních problémů, na které je možné narazit. Čtení ze souboru by mělo být vyhrazené pouze pro jedno vlákno. Zmíněnou kritickou sekci jsem ošetřil automatem ve třídě „FileInput", které zajistí, že čtení ze souboru může provádět pouze jedno vlákno.

2.2 Výstupní stavové soubory

Jelikož součástí zadání je bezprostřední zápis jednotlivých vláken do stavového souboru, je potřeba ošetřit možné kolize výstupů. Pro tento problém jsem využil mapu semaforů, která obsahuje semafor pro každý existující stavový soubor.

2.3 Přiřazení segmentů

Jednotlivé souborové segmenty jsou při chodu aplikace rozdělovány mezi zadaný počet vláken. Může však nastat problém přiřazení jednoho segmentu dvěma různým vláknům. Tuto skutečnost jsem ošetřil pomocí funkce `setAssigned`, která nastaví „flag"`isAssigned` na „true"a navrátí vláknu skutečnost jestli vlastní klíč ke zmíněnému segmentu.

3 Závěr

Po dokončení vývoje aplikace přišlo na řadu testování konfigurace. Všechny zkoušené výsledky jsem přehledně zapsal do tabulky.

Boss	Under-Boss	Master	Foreman	Slave	Potřebných sekund
1	1	1	1	1	6,5
4	1	1	1	1	5,8
1	4	1	1	1	5,9
1	1	4	1	1	6,0
1	1	1	4	1	6,2
1	1	1	1	4	7,9
4	3	2	1	1	5,7
3	3	1	1	1	5,6
5	20	10	10	10	24,0
5	5	10	10	30	37,5

Z této tabulky je možné vyčíst, že čím více spustíme vláken nižší úrovně, tím déle trvá chod programu. To je způsobené častějším křížením jednotlivých vláken a jejich postupné blokování. Nejlepší výsledek jsem získal při konfiguraci [3—3—1—1—1], kde chod aplikace trval 5,6 sekund. Naopak nejdelší zkoušený pokus při konfiguraci [5—5—10—10—30] trval 37,5 sekundy. U této konfigurace je jasně vidět, že více vláken rozhodně neznamená rychlejší chod aplikace.

Aplikace byla testována na počítači, který má procesor AMD Ryzen 5 o frekvenci 3.6 GHz, 16 GB operační paměti RAM, SSD disk a operační systém Ubuntu 19.10 Linux.