

Programovací seminář (IPS 22/23Z)

Úloha 2 - paralelní automaty

Paralelní automaty

Vytvořte program založený na vláknech, který bude provádět paralelní hledání podřetězců pomocí stavových automatů.

Výsledná binárka "aut" bude mít 2i+1 vstupních parametrů (příklady vstupu naleznete v souboru tests.txt):

```
aut MIN_SCORE STR1 SC1 [ STR2 SC2 ] [ STR3 SC3 ] ...
```

STRi: řetězec délky 5 nad abecedou {a-z,A-Z,0-9}. Pro jednoduchost uvažujte řetězce sestavené z 5-ti různých znaků.

SCi: score získané, pokud řádek na vstupu obsahuje podřetězec STRi.

MIN_SCORE: celé číslo, minimální score nutné pro výpis řádku (může být i záporné).

Minimální počet parametrů jsou 3, maximální není omezen (musí však být liché číslo).

Program aut vždy načte jednu řádku standardního vstupu a spočítá její score. Pokud score je alespoň

MIN_SCORE, tak je řádka vypsána na standardní výstup.

Výpočet score:

- 1) Počáteční score pro každý řádek je 0
- 2) Pro každý řetězec ověříme, zda-li řádka obsahuje podřetězec STRi.
- 3) Pokud ano, tak přičteme k aktuálnímu score hodnotu SCi (může být i záporná).

Příklad

Mějme následující vstup:

```
-----
```

Ahoj, tohle je cesta kolem Hradu.

Hradu zamavame: Ahoj1.

211 je kolem vysledku

```
-----
```

```
aut 3 "Ahoj1" 2 "kolem" 1 "Hradu" 3
```

Score jsou následující:

Ahoj, tohle je cesta kolem Hradu. --- score: 1+3=4

Hradu zamavame: Ahoj1. --- score: 3+2=5

211 je kolem vysledku. --- score: 1

--- score: 0

Vypsáno tedy bude:

```
-----
```

Ahoj, tohle je cesta kolem Hradu.

Hradu zamavame: Ahoj1.

```
-----
```

Výpočet bude probíhat následovně:

- 1) Spusí se "k" vláken, každé pro jeden řetězec STRi. Vlákna čekají, až jim bude předáno řízení hlavním vláknem.
 - 2) Hlavní vlákno načte jednu řádku ze vstupu.
 - 3) Pokud na vstupu žádná řádka nebyla, jsou všechna vlákna probuzena a korektně ukončena. Celý program končí.
 - 4) Řízení předáno vláknům pro testování zda-li načtená řádka obsahuje podřetězec STRi a vlákna __paralelně__ spočítají score.
- K výpočtu použijí stavový automat.
- 5) Řízení je předáno zpět hlavnímu vlákn.
 - 6) Pokud je skore alespoň zadané minimum, je řádka hlavním vláknem vypsána na standardní výstup.
 - 7) Pokračuje se bodem 2.

Požadavky na implementaci:

Program musí splňovat následující požadavky:

- 1) Pro každý řetězec STRi na vstupu se na začátku spustí jedno vlákno a to běží až do konce
- 2) V rámci jednoho vlákna lze provádět dotazy pouze na jeden podřetězec.
- 3) Pro dotazy na podřetězec implementujte stavový automat. Nevyužívejte knihovní funkce.
- 4) Pro komunikaci mezi vlákny využijte globální proměnná a pro synchronizaci semaforey (pthread mutex, c11 mutex, nebo C++ mutex).

5) Vlákna mohou aktivně čekat:

```
mutex.lock();
while (!podminka) {
    mutex.unlock();
    mutex.lock();
}
<nejaka prace>
mutex.unlock();
```

Uvažujme pouze ASCII znaky, kódování češtiny nebudeme využívat.

Dávejte pozor na data race (vstup, score, ...) a deadlock.

Není problém, aby více vláken najednou četlo ze stejné globální proměnné, ale pokud se zapisuje, je třeba zajistit atomicitu.

Za účelem testování můžete vložit na vhodná místa do kódu usleep(1). Např. před změnou score, načtení další řádky a podobně. V odevzdaném kódu již usleep nebude.

Implementace bude napsána v jazyce C s možností použití následujících komponent jazyka C++:

- vlákna a zámky z C++11 (#include <thread>, #include <mutex>)
- pole proměnné délky z C++11 (#include <vector>) pro uchování threadů a zámků.
- funkce int read_input_line(char *) ze souboru kostra_pgrep.c

Překlad bude prováděn buď:

```
$ gcc aut.c -o aut
```

nebo

```
$ g++ aut.c -o aut
```

s eventuálním parametrem -lpthread

Kostra programu a vzorové řešení

Na merlinovi v adresáři /usr/local/groups/verifikace/IPS-projekt2 naleznete následující soubory:

* aut - vzorové řešení

* tests.txt - testy

* kostra_aut.c - základní kostra programu, ze které můžete vyjít. Vytváří pole vláken a pole zámků a následně je ruší.

Odevzdání

Pracovat můžete ve dvoučlenných týmech. Každý z týmu pak odevzdá jeden soubor aut.c (přes IS VUT), kde bude v komentáři uvedeno jméno a login druhého člena týmu.

Na semináři dne 25.11. bude představeno vzorové řešení.

Naposledy změněno: Středa, 9. listopadu 2022, 09:22