

Programowanie współbieżne

Lista zadań nr 13

Na ćwiczenia 27. stycznia 2022

Zadanie 1. Wyjaśnij, jak działa metoda **exchange()** z klasy **LockFreeExchanger**. Dla każdego wywołania metody **compareAndSet()** wymień wszystkie powody, dla których może ono zawieść i wyjaśnij, jakie akcje podejmuje **exchange()** w takim przypadku. Dlaczego w pewnych dwóch miejscach w kodzie ustawia się zmienną **slot** przy pomocy **set()** a nie **compareAndSet()**?

Wskazówka: TAoMP2e, r. 11.4.1

Zadanie 2. Omów działanie klasy **EliminationBackoffStack**. Jaka rolę pełni w niej klasa **RangePolicy**? Wskaż punkty linearyzacji metod **push()** i **pop()**.

Wskazówka: TAoMP2e, r. 11.4

Zadanie 3. Opisz w sposób wysokopoziomowy (tzn. bez odwoływania się do kodu) algorytm implementowany przez metodę **getAndIncrement()** klasy **CombiningTree**.

Wskazówka: TAoMP2e, r. 12.3

Zadanie 4. Wyjaśnij rolę pełnioną przez flagę **locked** w klasie **Node** wykorzystywanej przez **CombiningTree**. W szczególności odpowiedz na pytania:

1. Dlaczego w rozpoczynając metody **precombine()** i **combine()** wątek wywołujący oczekuje na zgaszenie tej flagi (ustawienie na **false**)?
2. Dlaczego nie ma to miejsca w metodach **op()** i **distribute()**?
3. Dla każdego przypisania wartości zmiennej **locked** w metodach **precombine()**, **combine()**, **op()** i **distribute()** wyjaśnij, jaki wpływ ma to na pozostałe wątki wykonujące **getAndIncrement()**.

Uwaga: Rozwiązaniem każdego poniższych zadań powinien być działający kod wraz z przykładowym programem testowym.

Zadanie 5. Zadanie 12.2 z TAoMP2e.

Zadanie 6. Zadanie 12.3 z TAoMP2e.

Zadanie 7. Zadanie 12.4 z TAoMP2e.