

1. Вступ

Сучасні комп'ютерні системи ґрунтуються на використанні багатоядерних процесорів, кількість ядер у яких безперервно зростає. На сьогодні стандартом є восьмиядерний процесор, масовими стають 12-ядерні процесори, на підході 32 і 64-ядерні процесори, вже представлені на ринку.

Використання систем з різною кількістю ядер ставить завдання розробки програмного забезпечення, яке ефективно виконуватиметься в системах з різною кількістю ядер (масштабованих системах).

Сучасні мови та бібліотеки паралельного програмування забезпечують різні засоби роботи з потоками [1-5], існуючі засоби удосконалюються, а також з'являються нові. Вибір засобів програмування потоків та засобів організації їх взаємодії багато в чому визначає ефективність паралельної програми для масштабованої комп'ютерної системи.

2. Методологія

У публікації розкрито тему того що таке багатопоточність, як вона виглядає та за що відповідає, як організується багатопоточність, ті які на неї покладаються завдання.

3. Результати

Організація взаємодії потоків

Взаємодія потоків включає комунікацію (передачу даних між потоками) та синхронізацію потоків.

Реалізація цих дій залежить від моделі, обраної в мові для взаємодії:

- моделі, що базується на загальних змінних (shared variables model),
- моделі, що базується на повідомленнях (message passing model).

Більшість мов (бібліотек) орієнтовані на одну із зазначених моделей (як правило, першу), але є й такі, що підтримують обидві моделі

Завдання взаємного виключення

Вирішення завдання взаємного виключення передбачає контроль потоків при зверненні до загального ресурсу або контроль безпосередньо загального ресурсу [3, 5]. Перший вид контролю забезпечують семафори, мютекси, критичні секції, другий - монітори.

Завдання синхронізації

Завдання синхронізації передбачає синхронізацію двох потоків (один потік чекає на подію в іншому), а також колективну синхронізацію (один потік чекає на події в декількох потоках, кілька потоків чекають на подію в одному потоку, група потоків чекає на події в іншій групі потоків).

4. Ключові інсайти

Потоки – це послідовність даних або інструкцій, які можуть бути оброблені програмою або пристроєм. У багатьох випадках, потік може бути розглянутий як послідовність байтів, які переміщуються від одного місця до іншого.

Багатопоточність - це властивість багатьох сучасних операційних систем і мов програмування, яка дозволяє програмам виконувати кілька потоків в одному процесі. Це дозволяє програмі виконувати кілька завдань паралельно, що може призвести до покращення продуктивності та використання ресурсів.

5. Висновок

1. Сучасні бібліотеки та мови паралельного програмування забезпечують широкий спектр інструментів для роботи з потоками у паралельних системах.
2. Масштабовані паралельні програми характеризуються збільшенням та динамічною зміною кількості потоків у системі, що значно ускладнює організацію їх взаємодії.
3. Класичні низькорівневі механізми типу семафорів, мютексів, подій, а також критичних секції не повною мірою дозволяють організувати вирішення завдань синхронізації та взаємного виключення в системах, що масштабуються.
4. Для організації взаємодії потоків у системах, що масштабуються, найбільший інтерес представляють засоби, що реалізують концепцію моніторів, зокрема, механізм захищених модулів мови Ада, який також дозволяє об'єднати вирішення завдань взаємного виключення та синхронізації.