Питання до 5-2 лекції

ПІБ Гапей Максим Юрійович група ПД-31

Вписати від повідь на питання

|  |  |  |
| --- | --- | --- |
|  | Питання |  |
|  | Як перекладається Lock? | Блокування |
|  | Що таке блокування (Lock) або взаємне блокування в операційної системі? | Блокує та звільняє секцію. |
|  | Наведіть приклад взаємного блокування | Коли ми займаємо секцію, то ми чекаємо звільнення. Або якщо секція вільна, переводимо його в зайнятий |
|  | Які види блокування існують в операційних системах? | - зайнятий  - вільний |
|  | Що таке блокування виду зайнятий? | Секція очікує звільнення |
|  | Що таке блокування виду вільний? | Переведення в вільний стан |
|  | Які операції виконуються під час блокування в операційних системах? | - зайняти  - звільнити |
|  | Що таке операція блокування виду зайняти (перевести в стан "зайнято" або чекати звільнення)? | Блокується секція на вхід іншим ресурсам |
|  | Що таке операція блокування виду звільнити (перевести в стан "вільно")? | Процес виходить з секції і вона знову стає вільною |
|  | Які етапи здійснюються під час блокування? | Блокується секція та процес на зміну. |
|  | Що таке Критична секція? | Секція, де опрацьовуються процес |
|  | Що таке зайняти критичну секцію? | Процес входить в критичну секцію та блокує секцію і себе |
|  | Для чого використовується блокування? | - захисту даних  - процеси можуть займати різні блокування  - потрібно використовувати одне блокування для одного усіх критичних секцій, використовуваних одними і ті ж самі ресурси. |
|  | Чому блокування захищає дані ? | Тому що вона не перетворює дані в сміття. |
|  | Можуть чи ні процеси або потоки одночасно займати різні блокіровки? | Так, можуть. |
|  | Чому блокування захищає можливість процесів (потоків) займати різні блокування? | Для того щоб прогарам мала можливість використовувати декілька блокіровок |
|  | Як блокування захищає всі критичні секції від використання одних і тих же ресурсів однім блокуванням? | Щоб мати можливість використовувати один і той самий замок в різних ситуаціях |
|  | Можна змінювати чи ні баланс (втручатися в процес, що займає критичну секцію) під час зміни в критичній секції? | Ні |
|  | Що робіть замок відносно процесів в критичній секції? | Процеси не зміняться |
|  | Можна чи ні зайняти замок, що зайнятий? | Ні |
|  | Що нам потрібно в критичній секції? | - взаємні виключення  - прогрес  - конечні очікування  - продуктивність  - «справедливість» |
|  | Що таке Взаємне виключення в критичній секції? | Щоб в секції міг знаходитися тільки один процес |
|  | Що таке Прогрес в критичній секції? | Щоб процес поза критичної секції не мав можливості заборонити іншому процесу потрапити в критичну секцію |
|  | Що таке Кінцеве очікування в критичній секції? | Щоб ми розуміли, що процес вийде з критичної секції |
|  | Що таке Продуктивність в критичній секції? | Щоб була гарна продуктивність |
|  | Що таке Справедливість в критичній секції? | Щоб різні процеси мали можливість потрапити к критичну секцію |
|  | Що буде, якщо два процеси одночасно отримають інформацію про вільність одного і того ж замку? | Вони одночасно займають цей замок, та для кожного процеса замок стає зайнятим |
|  | Якім чином вирішується конфлікт зайнятості замка в операційної системі? | Завдяки операції TEST-AND-SET |
|  | Що таке TEST-AND-SET? | Операція на рівні заліза |
|  | Що таке флаг в TEST-AND-SET? | Бінарні змінна |
|  | Що таке семафор? | Цілочисельний лічильник, який обмежує кількість процесів, які можуть увійти в певну ділянку коду |
|  | Для чого потрібний семафор? | Щоб включати декілька процесів в себе |
|  | Які стани має семафор? | - ініціалізація  - збільшення  - зменшення |
|  | Що таке ініціалізація семафору? | Створення семафору з лічильником |
|  | Що таке збільшення (signal) семафору? | Збільшення лічильника |
|  | Що таке зменшення (wait) семафору? | Зменшення лічильника |
|  | Що таке блокування семафору? | Вихід з семафору |
|  | Опишіть зміст роботи семафору | Лічильник вказує нам, скільки потоків одночасно можуть отримувати доступ до загального ресурсу. |
|  | Коли семафорне не потрібний? | Коли є можливість обійтися замком |
|  | Недолік семафорів? | - процес може очікувати якоїсь події  - семафори ускладнюють програмування  - Інколи потрібно використовувати декілька семафорів |
|  | Може чи ні бути кілька семафорів в одному завданні? | Так |
|  | Що таке монітор в операційної системі? | Це семафор для користувачів |
|  | Як працює монітор операційної системі? | Є якісь данні та операції над цими даними |
|  | Переваги монітору з операційної системі? | - взаємне виключення  - умови блокування |
|  | Що таке Starvation? | Ресурсне голодування |
|  | Що таке Deadlock? | Взаємна блокіровка |
|  | Що таке Livelock? | Динамічний тупик |
|  | Що таке Взаємне блокування (deadlock) (або зависання)? | Коли два і більше потоків вічно очікують один одного |
|  | Що таке динамічний тупик (Livelock)? | Коли умови, за якими вони намагаються продовжити свою роботу, не можуь виконатися. |
|  | Що таке «взаємне виключення», М'ютекс (англ. Mutex, від mutual exclusion)? | Щоб в секції міг знаходитися тільки один процес |
|  | Що таке Стан гонки (race condition)? | потоки ділять між собою деякий ресурс, що не передбачає коректну роботу в такому випадку. |
|  | Що таке ресурсне голодування (Resource Starvation) або (starvation)? | Нестача ресурсів на всіх. |
|  | Умови виникнення Взаємного блокування (deadlock)? | - Взаємне виключення  - hold and wait  - система не може «забирати» ресурси у процеса  - циклічне очікування |
|  | Що робити з deadlock? | - Вирішення проблем  - Превентивної роботи  - Заборонити hold-and-wait  - Заборона циклічно блокування |
|  | В чому зміст Вирішення проблем щодо deadlock? | вирішувати проблемі в міру їх появи |
|  | В чому зміст Превентивної роботи щодо deadlock? | Зробити так щоб проблема не могла виникнути |
|  | В чому зміст Заборонити hold-and-wait (або переглянути  блокування) щодо deadlock? | заборона очікування, щоб не виникали deadlock |
|  | В чому зміст Заборона циклічно блокування (рівні,  доступу до блокірсасам) щодо deadlock? | Різні рівні доступа |