

Лабораторна робота № 5

Паралельне багатопотокове програмування на базі сучасних фреймворків

Вказівки до роботи: Завдання потрібно виконати з використанням багатопотокового програмування (на мовах C#, Kotlin, Java, C++, ...). Написати дві версії класів програми. У першій реалізувати синхронізацію потоків за допомогою *семфорів*, у другій — *моніторів*. Написати програму, яка моделює паралельне функціонування об'єктів предметної області.

1. Розв'язати задачу про виробників-споживачів у випадку одного споживача та n виробників, які можуть використовувати буфер розміру k . Споживач має зчитати усі значення, які йому передають виробники.
2. *Трамвай*. Трамвай, у який може вміститися n пасажирів, рухається по циклічному маршруту з k зупинками, на яких входять та виходять пасажирів. Реалізувати об'єкти «трамвай» та «пасажир» і змоделювати процес руху.
3. *Принтери*. Нехай n користувачів спільно та багаторазово використовують два принтери. Перед використанням принтера користувачі викликають функцію `request`. Ця функція чекає, поки один з двох принтерів не звільниться, і повертає ідентифікатор вільного принтера. Після використання принтера користувач звільняє його, викликаючи функцію `release`.
4. *Курник*. Є n пташенят та мама-квочка. Пташенята їдять із загальної миски, яка вміщує F порцій їжі. Кожне пташеня з'їдає порцію їжі, спить деякий час, а потім знову їсть. Коли закінчується їжа, то повідомляється квочка, яка наповнює миску F новими порціями їжі. Далі ці дії повторюються. Реалізувати класи «курник», «курча» та «квочка» і змоделювати функціонування курника.
5. *Ліфт*. Ліфт, у який може вміститися m пасажирів, стоїть на 1-му поверсі. Пасажири викликають ліфт та входять (якщо є вільні місця) і виходять з нього на потрібному їм поверсі. Реалізувати класи «ліфт» та «пасажир» і змоделювати процес руху та посадки/висадки пасажирів.

6. *Машини на мосту.* До вузького мосту під'їжджають машини з півночі та півдня. Машини, які рухаються у одному напрямку, можуть долати міст одночасно, а в протилежних — ні. Змоделювати процес руху.
7. *Обід філософів.* П'ять філософів сидять біля круглого столу. Вони проводять життя, чергуючи прийоми їжі та роздуми. У центрі столу знаходиться велике блюдо спагеті. У процесі їжі філософи повинні користуватися двома виделками. На жаль, їм дали всього п'ять виделок. Між кожною парою філософів лежить одна виделка і вони домовилися, що кожен буде користуватися тільки тими виделками, які лежать поруч з ним (зліва і справа). Завдання — написати програму, що моделює поведінку філософів. Програма повинна уникати ситуації, в якій всі філософи голодні, але жоден з них не може взяти обидві виделки — наприклад, коли кожен з них тримає по одній вилиці і не хоче віддавати її.
8. *Вулик.* Є n бджіл та 1 ведмідь. Вони використовують один горщик, який уміщує N порцій меду. Спочатку горщик порожній. Поки горщик не наповниться, ведмідь спить, потім з'їдає увесь мед та засипає. Кожна бджола (багаторазово) збирає одну порцію меду та кладе її у горщик. Бджола, яка приносить останню порцію меду та заповнює горщик, будить ведмедя.
9. *Банківський рахунок.* Кілька людей (потоків) використовують спільний рахунок. Можна розміщати або знімати кошти з рахунку. Поточний баланс рівний сумі усіх вкладених грошей мінус сума знятих коштів. Баланс не може бути від'ємним. Розміщати кошти можна без затримки, при вилученні коштів можливо є пауза, поки на рахунку не буде достатньої суми. Реалізувати два методи: `deposit(amount)` та `withdraw(amount)`.
10. *Американські гірки.* Є n потоків-пасажирів і один потік-вагончик. Пасажири чекають черги проїхати в вагончику, який вміщує C людей, $C < n$. Вагончик може їхати тільки заповненим. Напишіть коди потоків-пасажирів і потоку-вагончика і розробіть засоби для їх синхронізації. Реалізувати три операції: `takeRide`, яку викликають пасажири, `load` і `unload`, які викликає потік-вагончик.