

НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ «КПІ»

ФАКУЛЬТЕТ ПРИКЛАДНОЇ МАТЕМАТИКИ

Кафедра спеціалізованих комп'ютерних систем

ЛАБОРАТОРНА РОБОТА №5

з дисципліни

“Паралельні та розподілені обчислення”

ТЕМА: «Засоби взаємодії паралельних потоків мови Java»

Підготував: доц. Марченко О.І.

Copyright © 2009 – 2016, Марченко О.І.

Київ – 2009-2016

Постановка завдання та вимоги до виконання програми

1. Написати програму на мові Java, яка реалізує роботу паралельних потоків згідно заданої за варіантом схеми. Особливості реалізації синхронізації паралельних потоків та взаємного виключення потоків при доступі до спільних ресурсів задані за варіантами у таблиці завдань.
2. При написанні програми виконати повне трасування роботи програми за допомогою операторів друку, тобто розставити в програмі оператори друку таким чином, щоб можна було прослідкувати всі варіанти виконання паралельних потоків і впевнитись у коректності роботи програми. Протокол трасування рекомендується записувати у файл (log-файл).
3. Запуск усіх потоків повинен бути виконаний у головній програмі.
4. Кожен потік повинен бути організованим у вигляді нескінченного циклу.
5. Всі дії задані за варіантами, що вказані у таблиці, повинні бути виконані всередині цього нескінченного циклу.
6. Взаємне розташування операторів синхронізації та доступу до спільного ресурсу, якщо вони знаходяться у одному потоці, є довільним.
7. Оскільки синхронізація за допомогою семафорів Sem1, Sem2 згідно завдання розташована всередині нескінченних циклів, то відразу після виконання синхронізації ці семафори повинні бути знову встановлені у початковий закритий стан.
8. Закінчення програми можна виконати двома способами:
 - примусовим перериванням за допомогою натиснення комбінації клавіш Ctrl+C;
 - оператором виходу з циклу break при виконанні умови, яка стає істинною, коли буфер спільного ресурсу повністю заповнюється і повністю звільняється мінімум по два рази.
9. Якщо при реалізації паралельних потоків була використаний метод sleep(), то передбачити режим запуску програми з «відключеними» викликами sleep().
10. Виконати налагодження написаної програми.

Зміст звіту

1. Загальна постановка завдання.
2. Завдання конкретного варіанту.
3. Текст програми.
4. Декілька протоколів роботи програми, які демонструють різні випадки роботи паралельних потоків.

Контрольні питання

Знати принципи організації і засоби комунікації та синхронізації паралельних потоків із загальнотеоретичної точки зору, а також реалізацію цих засобів у мові Java.

Пояснення до таблиці з варіантами завдань

1. Потоки P1 – P5 повинні бути організовані у вигляді нескінченних циклів, в тілі яких повинні бути реалізовані всі дії, які задані за варіантами завдань.
2. **CR1** □ перший спільний ресурс (common resource) у вигляді буфера для обміну даними між потоками-постачальниками і потоками-споживачами P1 – P5. Спосіб реалізації буфера визначається у таблиці 1 за варіантами та роль кожного з потоків P1 – P5.
Взаємне виключення потоків P1 – P5 при доступі до CR1 повинно бути реалізоване за допомогою монітору. В точках читання/запису інформації спільного ресурсу CR1 потрібно виконати доступ до CR1 за допомогою синхронізованих методів монітора, вбудованого у клас цього спільного ресурсу. Вид доступу (запис чи читання) визначається типом потоку – Постачальник чи Споживач.
3. **CR2** □ другий спільний ресурс (common resource), який складається з неатомарних змінних (по одній змінній кожного з елементарних типів мови Java). В таблиці варіантів зазначено у яких потоках повинен бути виконаний доступ до CR2. Під час кожного доступу до CR2 повинно бути виконано:
 - а) зміна значень 4-х довільних змінних (у різних потоках повинні змінюватись значення різних змінних спільного ресурсу CR2);
 - б) виведення на у файл протоколу роботи програми (log-файл) поточних значень всіх змінних CR2.
 - в) Ініціалізацію змінних CR2 виконати при створенні об'єкту класу CR2.**Взаємне виключення заданих за варіантом потоків від одночасного доступу до змінних CR2 виконати за допомогою м'ютекса.**
4. В тілі нескінченних циклів потоків, що вказані у колонці таблиці «Двійкові семафори Sem1 та Sem2», повинна бути виконана **повна синхронізація за допомогою двох двійкових семафорів з ідентифікаторами Sem1 та Sem2**
5. В тілі нескінченних циклів потоків, що вказані у колонці таблиці «Циклічні бар'єри CB1 та CB2», повинна бути виконана **в залежності від варіанту або одночасна синхронізація трьох потоків за допомогою або одного циклічного бар'єру (Cyclic Barrier) CB1 або двох груп по два потоки за допомогою двох циклічних бар'єрів CB1 та CB2.**

Варіанти завдань

№ варіанту	Ім'я та тип процесів, у яких реалізований доступ до спільного ресурсу CR1 (Постачальник — П, Споживач — С)					Спільні ресурси CR1 та CR2		Засоби синхронізації та потоки, у яких вони повинні бути реалізовані дані засоби	
	P1	P2	P3	P4	P5	CR1 Структура даних, що використовується у якості CR1 (буферу обміну даними)	CR2 (Неатомарні змінні) Потоки, у яких повинні бути реалізовані читання та модифікація CR2	Двійкові семафори Sem1 та Sem2	Циклічні бар'єри CB1 та CB2
1	С	С	П	П	П	Стек у вигляді Вектора	P3, P4, P5	P1 та P2	CB1: P1, P2 CB2: P4, P5
2	П	П	П	П	С	Циклічний буфер у вигляді Вектора	P1, P2, P5	P3 та P4	CB1: P1, P2, P5
3	П	С	С	С	П	Циклічний буфер у вигляді Вектора	P1, P2, P3	P4 та P5	CB1: P1, P3 CB2: P2, P3
4	С	С	П	С	С	Стек у вигляді Вектора	P1, P3, P5	P2 та P4	CB1: P1, P4, P5
5	С	П	С	П	П	Циклічний буфер у вигляді Вектора	P1, P3, P4	P2 та P5	CB1: P2, P4 CB2: P1, P5
6	П	П	П	С	П	Стек у вигляді Вектора	P1, P2, P3	P1 та P4	CB1: P2, P3, P5
7	П	С	С	П	С	Стек у вигляді вектора	P1, P4, P5	P2 та P3	CB1: P1, P5 CB2: P3, P4
8	С	П	С	С	С	Циклічний буфер у вигляді вектора	P3, P4, P5	P1 та P2	CB1: P2, P4, P5
9	П	П	С	П	П	Циклічний буфер у вигляді вектора	P2, P3, P4	P1 та P5	CB1: P2, P4 CB2: P3, P4
10	С	П	П	С	П	Стек у вигляді вектора	P1, P2, P3	P4 та P5	CB1: P1, P2, P3
11	П	С	П	С	С	Циклічний буфер у вигляді вектора	P1, P3, P5	P2 та P4	CB1: P4, P5 CB2: P1, P3
12	С	П	П	П	С	Циклічний буфер у вигляді вектора	P1, P2, P5	P3 та P4	CB1: P1, P2, P5
13	П	С	С	С	С	Стек у вигляді вектора	P1, P3, P4	P2 та P5	CB1: P3, P4 CB2: P1, P4

14	С	С	С	П	С	Циклічний буфер у вигляді вектора	P1, P3, P5	P2 та P4	CB1: P1, P4, P5
15	П	С	П	П	П	Стек у вигляді вектора	P1, P2, P3	P4 та P5	CB1: P1, P3 CB2: P2, P4
16	П	П	С	С	С	Стек у вигляді вектора	P1, P2, P3	P4 та P5	CB1: P2, P3, P4
17	С	П	П	П	П	Циклічний буфер у вигляді вектора	P2, P4, P5	P1 та P3	CB1: P2, P3 CB2: P4, P5
18	П	С	С	П	П	Стек у вигляді вектора	P2, P3, P5	P1 та P4	CB1: P2, P3, P5
19	С	С	С	С	П	Стек у вигляді вектора	P1, P4, P5	P2 та P3	CB1: P1, P2 CB2: P3, P4
20	С	П	С	С	П	Циклічний буфер у вигляді вектора	P1, P2, P4	P3 та P5	CB1: P1, P4, P5
21	П	С	П	С	П	Циклічний буфер у вигляді вектора	P2, P4, P5	P1 та P3	CB1: P1, P2 CB2: P4, P5
22	П	С	П	П	П	Стек у вигляді вектора	P1, P2, P5	P3 та P4	CB1: P1, P3, P5
23	С	С	П	С	П	Стек у вигляді вектора	P1, P3, P4	P2 та P5	CB1: P1, P3 CB2: P3, P4
24	С	П	С	С	С	Циклічний буфер у вигляді вектора	P2, P3, P4	P1 та P5	CB1: P2, P3, P4