

Методичні вказівки до виконання навчальних проектів з курсу «Методи паралельних обчислень»

Навчальний проект №1. «Моделювання броунівського руху засобами мультипоточного програмування»

Опис прикладної задачі.

Розглядається процес випадкового блукання (броунівського руху) в одновимірному кристалі. Нехай є простір, що є кінцевим набором клітинок N , по яких розподілене деяка кількість K атомів домішок. У кожен момент часу атом може перейти в будь-яку з двох сусідніх кліток з одним і тим же порогом «вірогідності» p : $1 \geq p \geq 0$.

Для простоти вважаємо, що на краях кристалу атоми домішки відбиваються (тобто, залишаються на місці). Таким чином, загальна кількість частинок домішки залишається незмінною. Вважатимемо також, що у початковому стані всі атоми домішки зосередженні в крайній лівій клітинці.

Правила переходу частинки домішки полягають у наступному:

1. Кожен потік моделювання частинки у кожному акті моделювання має згенерувати випадкове число m в інтервалі $[0, 1]$. Якщо $m > p$, то перехід здійснюється вправо, в протилежному випадку – вліво.
2. Якщо частинка знаходиться в крайній позиції (зліва чи справа), то при спробі вийти за межі кристала відбувається «віддзеркалення» і частинка залишається на місці.

Завдання

1. Потрібно написати програму, що моделює поведінку частинок у одновимірному кристалі засобами мультипоточного програмування (мовами Java, C# або іншими – на вибір студента), використовуючи для кожного атома домішки свій потік виконання. Результатом роботи моделювання має бути набір «моментальних знімків», що характеризують стан розподілу частинок домішки по клітинах кристалу від початку до кінця терміну моделювання (взяти 1 хв.) з дискретністю кроку відображення стану 5 сек.
2. Провести дослідження, як впливає гранульованість блокувальних операцій (одна клітина чи увесь масив клітин кристала) на швидкодію мультипоточної програми моделювання.

Порядок виконання:

1. Вивчити організацію паралельних процесів як потоків виконання із спільною пам'яттю у вибраній мові програмування. Наприклад, у мові Java – це: 1) потоки

Threads і інтерфейс Runnable; 2) конструкція *synchronized* і її використання для забезпечення монопольного доступу до об'єкту.

- Далі послідовно виконати кроки написання програми (на прикладі застосування засобів Java):
- Кристал у програмі може відповідати масив цілих чисел `cells`, від 0 до `N`. Значення `cells[i]`-- кількість атомів домішки в i -тій клітинці кристала.
- Кожному атому домішки відповідає потік виконання, який містить в собі індекс атома домішки. Робота потоку полягає в модифікації масиву `cells` і зміни свого індексу в нескінченному циклі.
- Метод `main` ініціалізує масив, запускаючи потоки і чекає 10 секунд, після цього зупиняє всі потоки і закончує роботу.
- В кінці програми необхідне переобчислити загальну кількість атомів домішки (вона не повинна змінитися) і вивести його на екран.

Необхідно написати 2 версії цієї програми - одну некоректну, яка НЕ використовує блокування даних, іншу - коректну, яка використовує модифікатор *synchronized*.

Головні програми повинні називатися `Cells0` і `Cells1` відповідно.

У головних класах може бути визначений `public` метод `getCell(i)`, який повертає кількість атомів домішки в клітку i .

Параметрами методу `main` повинні бути N , K і p відповідно, розглянуті вище.