4.Анализ эффективности многопоточной обработки

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| M | N | 10 | 100 | 1000 | 100 тыс. | 1 млн. | 10 млн. | 1 млрд. |
| 1 | | 1 мс | 4 мс | 7 мс | 647 мс | 6594 мс | 70220мс | - |
| 2 | | 1 мс | 2 мс | 7 мс | 661 мс | 6744 мс | 70150мс | - |
| 3 | | 1 мс | 2 мс | 7 мс | 682 мс | 6949 мс | 71716мс | - |
| 4 | | 1 мс | 2 мс | 8 мс | 684 мс | 7005 мс | 72633мс | - |
| 5 | | 1 мс | 2 мс | 8 мс | 677 мс | 6978 мс | 70915мс | - |
| 10 | | 2 мс | 2 мс | 9 мс | 675 мс | 6832 мс | 68296мс | - |
| 20 | | 2 мс | 3 мс | 9 мс | 669 мс | 7266 мс | 70359мс | - |
| 30 | | 3 мс | 4 мс | 9 мс | 667 мс | 7180 мс | 70136мс | - |
| 100 | | 7 мс | 8 мс | 12 мс | 668 мс | 7167 мс | 70774мс | - |

Ускорение (10): 1/7 = 0,14

Эффективность:0,14/100=0,0014

Ускорение (100):4/8 = 0,5

Эффективность: 0,5/100 = 0,005

Ускорение (1000):7/12 = 0,58

Эффективность: 0,58/100 = 0,0058

Ускорение (100тыс.):647/668 = 0,97

Эффективность: 0,97/100 = 0,0097

Ускорение (1млн.): 6594/7167 = 0,92

Эффективность:0,92/100 = 0,0092

Ускорение (10млн.): 70220/70774 = 0,99

Эффективность:0,99/100 = 0.0099

5.Анализ эффективности многопоточной обработки при усложнении (возведение в степень)

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| M | N | 10 | 100 | 1000 | 100 тыс. | 1 млн. | 10 млн. | 1 млрд. |
| 1 | | 1 мс | 2 мс | 7 мс | 612 мс | 6525 мс | 66169мс | - |
| 2 | | 1 мс | 2 мс | 7 мс | 629 мс | 6852мс | 68486мс | - |
| 3 | | 1 мс | 2 мс | 8 мс | 670 мс | 7310 мс | 73621мс | - |
| 4 | | 1 мс | 2 мс | 7 мс | 686 мс | 7590 мс | 76058мс | - |
| 5 | | 1 мс | 2 мс | 7 мс | 701 мс | 7364 мс | 72635мс | - |
| 10 | | 2 мс | 2 мс | 8 мс | 668 мс | 7037 мс | 71868мс | - |
| 20 | | 2 мс | 3 мс | 9 мс | 681 мс | 6809 мс | 69406мс | - |
| 30 | | 3 мс | 3 мс | 10 мс | 669 мс | 6923 мс | 69058мс | - |
| 100 | | 5 мс | 6 мс | 12 мс | 676 мс | 7056мс | 71009мс | - |

Ускорение (10): 1/5 = 0,2

Эффективность:0,2/100=0,002

Ускорение (100):2/6 = 0,33

Эффективность: 0,0033/100 = 0,0033

Ускорение (1000):7/12 = 0,58

Эффективность: 0,58/100 = 0,0058

Ускорение (100тыс.):612/676 = 0,91

Эффективность: 0,91/100 = 0,0091

Ускорение (1млн.): 6525/7056 = 0,92

Эффективность:0,92/100 = 0,0092

Ускорение (10млн.): 66169/71009 = 0,93

Эффективность:0,93/100 = 0.0093

6. При таком неравномерном разделении ресурсы будут нерационально использоваться. Первый поток быстро завершит свою работу и будет простаивать, в то время как второй поток будет работать на полную мощность, возможно приводя к перегрузке процессора или других ресурсов.

**Вопросы:**

1.Почему эффект от распараллеливания наблюдается только при большем числе элементов?

Накладные расходы на управление потоками: Создание, управление и синхронизация потоков требует определенные накладные расходы на ресурсы процессора и память. Эти накладные расходы могут превышать выгоду от параллельной обработки для меньших объемов данных.

Потенциал для параллельной обработки: Многие аппаратные ресурсы, такие как многоядерные процессоры и графические процессоры, имеют потенциал для параллельной обработки больших объемов данных. Поэтому, при большем числе элементов, есть больше возможностей для эффективной параллельной обработки.

Оптимизация и уровень параллелизма: Некоторые алгоритмы и структуры данных могут быть оптимизированы для параллельной обработки больших объемов данных. Например, использование разделения данных и слияния результатов может быть более эффективным при больших объемах данных.

2.Как влияет увеличение сложности обработки на эффективность многопоточной обработки?

Распределение нагрузки: Сложные операции могут быть разделены между несколькими потоками, что позволяет распределить нагрузку на несколько ядер процессора или даже на несколько физических процессоров. Это может ускорить общую обработку данных.

Время на ожидание: Во время выполнения сложных операций один поток может ожидать выполнения операции ввода-вывода или другого длительного процесса. В случае многопоточной обработки другие потоки могут продолжать работу в это время, что повышает использование ресурсов процессора.

3.Какое число потоков является оптимальным для конкретной вычислительной системы? Как его подобрать?

Используйте количество потоков, соответствующее физическим ядрам процессора. Это обычно является хорошим начальным значением, поскольку физические ядра могут обрабатывать потоки параллельно без дополнительных накладных расходов.

Размер кэша процессора. Увеличение количества потоков может привести к конкуренции за ресурсы, такие как кэш процессора. Таким образом, оптимальное количество потоков также зависит от характеристик кэша процессора.

Производите эксперименты. Для нахождения оптимального количества потоков лучше всего провести ряд экспериментов. Можно попробовать разное количество потоков и измерить время выполнения для определения оптимального значения.

4.Почему неравномерность загрузки потоков приводит к снижению эффективности многопоточной обработки?

Простаивающие потоки: Если некоторые потоки завершили свою работу раньше других из-за неравномерности нагрузки, они могут простаивать, ждать результатов других потоков или просто занимать ресурсы процессора без какого-либо активного вклада. Это приводит к потере производительности и неэффективному использованию ресурсов.

Конкуренция за ресурсы: Если одни потоки завершают свою работу быстрее других, они могут конкурировать за доступ к общим ресурсам, таким как память или кэш процессора. Это может привести к увеличению времени ожидания и снижению производительности.

Динамическое перераспределение: Некоторые системы управления потоками могут пытаться перераспределить работу с неактивных потоков на более загруженные, но это также влечет за собой накладные расходы и может снизить общую производительность.

5.Как логичнее всего реализовать обработку таких данных

Разделение данных: Разделите данные на равные или пропорциональные части, которые могут быть обработаны независимо друг от друга. Это позволит каждому потоку обрабатывать свою часть данных, минимизируя конкуренцию за ресурсы.

Синхронизация и координация: Используйте механизмы синхронизации, такие как мьютексы, для координации работы потоков и предотвращения возможных конфликтов при доступе к общим данным.