**Лабораторная работа №4**

**Критические секции и атомарные операции**

**Цель:** изучить основные особенности использования критических секций и атомарных операций в OpenMP на примере использования в рамках языка С++.

Для получения **теоретических сведений** настоятельно рекомендуется при домашней подготовке изучить материалы, представленные в списке литературы в конце разработки, а также прочие материалы по тематике лабораторной работы, представленные в открытых источниках.

Далее следует краткий конспект материала, приведенного в данных источниках, в конце включающий короткие примеры фрагментов программ.

###### 1. Обеспечение атомарности (неделимости) операций

Действие над общей переменной может быть выполнено как атомарная (неделимая) операция при помощи директивы atomic. Формат директивы имеет вид:

#pragma omp atomic <expression>

где expression должно иметь вид:

x++; или ++x; или x--; или --x;

где x есть имя любой целой скалярной переменной. Директива atomic может быть записана и в виде:

#pragma omp atomic

x <operator>= <expression>

где x есть имя скалярной переменной, выражение expression не должно включать переменную x, а допустимыми значениями для поля operator являются следующие операции (которые не могут быть перегружены):

+, \*, -, /, &, |, ^, >>, <<

Как следует из названия, операция директивы atomic выполняется как неделимое действие над указанной общей переменной, и, как результат, никакие другие потоки не могут получить доступ к этой переменной в этот момент времени.

###### 2. Использование критических секций

Действия над общими переменными могут быть организованы в виде критической секции, т.е. как блок программного кода, который может выполняться только одним потоком в каждый конкретный момент времени. При попытке входа в критическую секцию, которая уже исполняется одним из потоков, все другие потоки приостанавливаются (блокируются). Как только критическая секция освобождается, один из приостановленных потоков (если они имеются) активизируется для выполнения критической секции.

Определение критической секции в OpenMP осуществляется при помощи директивы critical, формат записи которой имеет вид:

#pragma omp critical [(name)] <block>

Как можно заметить, критические секции могут быть именованными – можно рекомендовать активное использование данной возможности для разделения критических секций, т. к. это позволит уменьшить число блокировок процессов.

**Примеры**

Пример 1. Использование директивы atomic

total = 0;

#pragma omp parallel shared(**total**,a) private(i,j,sum)

{

#pragma omp for

for (i=0; i < NMAX; i++)

{

sum = 0;

for (j=0; j < NMAX; j++)

sum += a[i][j];

printf ("Сумма элементов строки %d равна %f\n",i,sum);

**#pragma omp atomic**

total += sum;

}

} */\* Завершение параллельного фрагмента \*/*

printf ("Сумма элементов матрицы равна %f\n",total);

Пример 2. Использование критических секций

smax = 0;

#pragma omp parallel shared(smax,a) private(i,j,sum)

{

#pragma omp for

for (i=0; i < NMAX; i++)

{

sum = 0;

for (j=0; j < NMAX; j++)

sum += a[i][j];

printf ("Сумма элементов строки %d равна %f\n",i,sum);

if ( sum > smax )

**#pragma omp critical**

if ( sum > smax )

smax = sum;

}

} */\* Завершение параллельного фрагмента \*/*

printf ("Максимальная сумма равна %f\n",smax);

**Лабораторные задания**

**Задание.** Модифицировать программу, составленную по Заданию 1 в Л.Р. №3, используя синхронизацию вместо редукции. Измерять время работы программы для тех же значений параметров, что были использованы при выполнении Л.Р. №3. **Результаты сравнить и занести в отчёт.**

**Контрольные вопросы**

1. Что понимается под атомарной (неделимой) операцией?
2. Как определяется критическая секция?

**Требования к сдаче работы**

1. При домашней подготовке изучить теоретический материал по тематике лабораторной работы, представленный в списке литературы ниже, выполнить представленные примеры, занести в отчёт результаты выполнения.
2. Продемонстрировать выполнение лабораторных заданий.
3. Ответить на контрольные вопросы.
4. Показать преподавателю отчет.

**Литература**

1. Спецификации стандарта OpenMP (на английском языке):

<http://openmp.org/wp/openmp-specifications/>

2. Материалы, представленные на сайте intuit.ru в рамках курса «Intel Parallel Programming Professional (Introduction)»:

<http://old.intuit.ru/department/supercomputing/ppinteltt/4/>

3. С.А. Лупин, М.А. Посыпкин Технологии параллельного программирования. – М.: ИД «ФОРУМ»: ИНФРА-М, 2011. – С. 119-145. *(Глава, посвященная OpenMP)*

4. Канг Су Гэтлин, Пит Айсенси - OpenMP и C++:

<http://msdn.microsoft.com/ru-ru/library/dd335940.aspx>

5. API OpenMP C и C++

<http://msdn.microsoft.com/ru-ru/library/8y6825x5.aspx>