

<https://github.com/itiscs/MPI>

Онлайн-компилятор

<https://www.codingame.com/playgrounds/54443/openmp/hello-openmp>

Общие и защищенные переменные. Разграничения доступа к переменным, конструкции разделения работ не итерационного типа. (10 баллов)

1. Написать программу где каждый поток печатает свой идентификатор, количество потоков всего и строку «Hello World». Запустить программу с 8 потоками. Всегда ли вывод идентичен? Почему? (1 балл)
2. Написать программу, в которой определить две параллельные области, выполнение которых зависит от условного оператора **#pragma omp parallel if(...)**, если заданное значение числа нитей больше 2, область выполняется параллельно, иначе не параллельно. Число нитей перед первой областью задать равным 3, перед второй – равным 2. Внутри параллельных областей определить количество нитей и номер каждой нити, результат выдать на экран. Убедиться в правильности работы программы. (2 балла)
3. Написать программу, в которой объявить и присвоить начальные значения целочисленным переменным: **a** и **b**, определить две параллельные области, **первая** – с числом **нитей 2**, **вторая** – с числом **нитей 4**. Для **первой** области переменные **a** и **b** объявить защищенными, с режимом доступа **private** и **firstprivate** соответственно, внутри области определить номер нити и выполнить увеличение значения переменных на эту величину. Выдать на экран значения переменных до входа в параллельную область, внутри параллельной области и после выхода из параллельной области. Для **второй** области переменную **a** объявить общей, переменную **b** объявить защищенной, с режимом доступа **private**, определить номер нити и выполнить уменьшение значения переменных на эту величину. Выдать на экран значения переменных до входа в параллельную область, внутри параллельной области и после выхода из параллельной области. Неинициализированные переменные инициализировать. (2 балла)
4. Написать программу, в которой объявить и присвоить начальные значения целочисленным массивам **a[10]** и **b[10]**, определить параллельную область, количество нитей задать равным 2, выделить код для основной (номер 0) и нити с номером 1. Основная нить (**master**) должна выполнять поиск **min** значения элементов массива **a**, нить с номером 1 - поиск **max** значения элементов массива **b**. Результат выдать на экран. (2 балла)
5. Написать программу, в которой объявить и присвоить начальные значения элементам двумерного массива **d[6][8]**, для инициализации значений использовать генератор случайных чисел. Используя конструкцию директивы **sections...section** определить три секции для выполнения следующих операций:
первая секция выполняет вычисление среднего арифметического значения элементов двумерного массива,
вторая секция выполняет вычисление минимального и максимального значений элементов двумерного массива,
третья секция выполняет вычисление количества элементов массива, числовые значения которых кратны 3.
В каждой секции определить и выдать на экран **номер исполняющей нити и результат выполнения вычислений**. (3 балла)

Конструкции разделения работ итерационного типа. (11 баллов)

6. Написать программу, в которой объявить и присвоить начальные значения целочисленному массиву **a[100]**. Используя конструкцию **parallel for** и **reduction** вычислить средние арифметические значения элементов массива **a**. Одно при помощи **reduction**, другое без. Сравнить полученные значения. Объяснить возможную разность результатов. (2 балла)

7. Написать программу, в которой определить две параллельные области, каждая из которых содержит итерационную конструкцию **for** выполняющую инициализацию элементов одномерных массивов целых чисел **a[12]**, **b[12]** и **c[12]**. Число нитей перед первой областью задать равным 3, перед второй – равным 4. Первая параллельная область выполняет инициализацию элементов массивов **a** и **b** с использованием **статического распределения** итераций, размер порции итераций выбрать самостоятельно, вторая параллельная область выполняет инициализацию элементов массива **c** по следующему правилу $c[i] = a[i] + b[i]$, с использованием **динамического распределения** итераций, размер порции итераций выбрать самостоятельно. В каждой области определить и выдать на экран количество нитей, номер нити и результат выполнения цикла. Убедиться в правильности работы программы. (2 баллов)

8. Написать программу, в которой объявлен массив из 16000 элементов и инициализирован так, что значение элемента массива равно его порядковому номеру. Затем создайте результирующий массив, в котором (за исключением крайних элементов) будут средние значения исходного массива:

$$b[i] = (a[i-1] + a[i] + a[i+1])/3.0$$

Запустите программу с 8-ю процессами при различных типах распределения работ (static, dynamic, guided, auto(runtime, если auto не работает)) и **посмотреть время на разных размерах порций**. (4 балла)

9. Используя возможности OpenMP написать программу умножения матрицы на вектор. **Сравнить время выполнения** последовательной и параллельных программ (выбрать наилучший **schedule**). Определить размеры матрицы при которых параллельная программа начинает работать быстрее последовательной (3 балла)

Операции синхронизации работ (11 баллов)

10. Написать программу, в которой объявить и присвоить начальные значения элементам двумерного массива **d[6][8]**, для инициализации значений использовать генератор случайных чисел. Используя конструкцию директивы **omp parallel for** и **omp critical** определить минимальное и максимальное значения элементов двумерного массива. Реализовать аналог reduction для max и min. Количество нитей задать самостоятельно. Результат выдать на экран. (2 балла)

11. Написать программу, в которой объявить и присвоить начальные значения массиву целых чисел **a[30]**, для инициализации значений использовать генератор случайных чисел. Используя конструкцию **omp parallel for** и **omp atomic** вычислить количество элементов массива, числовые значения которых кратны 9. Количество нитей задать самостоятельно. Результат выдать на экран. (2 балла)

12. Написать программу, в которой, объявить и заполнить случайными значениями массив целых чисел. Используя возможности OpenMP найти максимальное значение элементов массива кратных 7. Длину массива и количество потоков определить самостоятельно. Результат выдать на экран. Для синхронизации числовых значений максимума используется механизм **omp_lock**. (2 балла)

13. Модифицируйте задачу 1 так, чтобы потоки распечатывали свои идентификаторы в обратном

(убывающем) порядке. Существует как минимум 5 способов решения. Постарайтесь найти как можно больше. (каждый вариант 1 балл – max 5 баллов)