

Задачи к экзамену по курсу «Параллельное программирование»

1. Написать параллельную программу для нахождения количества простых чисел в диапазоне от 1 до 10000, разделив этот диапазон равномерно между имеющимися процессами параллельной программы.
2. Написать параллельную программу для нахождения суммы всех целых чисел от 1 до 10000, разделив диапазон слагаемых равномерно между имеющимися процессорами.
3. Написать параллельную программу для поиска минимального элемента в массиве. Массив разделить равномерно на части между процессами параллельной программы, на каждом процессе найти локальный минимум, локальные минимумы переслать на нулевой процесс и выбрать при пересылке глобальный минимум.
4. Написать параллельную программу для поиска максимального элемента в массиве. Массив разделить равномерно на части между имеющимися процессами, на каждом процессе найти локальный максимум, локальные максимумы переслать на нулевой процесс и выбрать при пересылке глобальный максимум.
5. Организовать пересылку каждым процессом своего ранга процессу с рангом на единицу большим. Процесс с наибольшим рангом пересылает значение ранга нулевому процессу. Нулевой процесс выводит полученный ранг.
6. Написать параллельную программу для вычисления интеграла от заданной функции на заданном отрезке методом прямоугольников. Отрезок интегрирования разбить на равные части и вычислить на отдельных процессах интеграл методом прямоугольников на отдельных подотрезках. При вычислении интеграла на подотрезках использовать одинаковое число узлов. Собрать значения подинтегралов на нулевом процессе, при сборке найти полное значение интеграла.
7. Написать параллельную программу с двумя процессами для сортировки массива. Нулевой процесс получает исходный массив, делит его на две половины и отправляет вторую половину первому процессу. Каждый процесс сортирует свою половину и отправляет другому процессу. Каждый процесс сливает две отсортированные половины в единый упорядоченный массив.
8. Написать параллельную программу для нахождения минимального значения функции от одной переменной на заданном отрезке. Отрезок разделить на одинаковые части между процессами. Поиск минимума выполнять методом перебора, задав на всех процессах одинаковое число узлов, в которых вычисляются значения функции. Переслать локальные минимумы на нулевой процесс, при пересылке найти глобальный минимум.
9. Написать параллельную программу для нахождения минимального значения функции от одной переменной на заданном отрезке методом перебора, то есть путем вычисления значения функции в заданном числе равноотстоящих узлов и выбора среди вычисленных значений минимального. Исходными данными считать границы отрезка поиска минимума функции и число точек, в которых должна быть вычислена функция. Отрезок поиска разделить на одинаковые части между процессами. Поиск минимума выполнять методом перебора, задав на всех процессах одинаковое число узлов, в которых вычисляются значения функции. Каждый процесс должен определить локальный минимум функции на своем отрезке и значение аргумента, при котором функция имеет минимальное значение. На нулевом процессе установить значение глобального минимума и значение аргумента, при котором достигается глобальный минимум.
10. Написать параллельную программу для поиска минимального элемента в массиве. Массив разделить равномерно на части между процессами параллельной программы, на каждом

процессе найти локальный минимум и номер в массиве минимального элемента. На нулевом процессе определить глобальный минимум и номер элемента массива, с наименьшим значением.

11. Написать параллельную программу, которая ищет в каждой тысяче натуральных чисел количество простых чисел. Диапазоны натуральных чисел имеют вид: $1, 2, \dots, 1000$; $1001, 1001, \dots, 1999$, $2000; \dots; k1000 + 1, k1000 + 2, \dots, (k + 1)1000$, $k = 0, 1, 2, \dots$. Количество тысяч k задается путем ввода и равномерно распределяется между процессами программы. Программа должна вывести номер диапазона и количество простых чисел на этом диапазоне.
12. Написать параллельную MPI-программу поиска минимального элемента квадратной матрицы. Матрицу разделить на полосы по числу процессов программы. Каждый процесс отыскивает локально минимальный элемент в своей полосе, номер строки и номер столбца. Результаты поиска минимума собираются на нулевом процессе, определяется глобально минимальный элемент, его номер строки и столбца.
13. Написать многопоточную программу, использующую технологию OpenMp, для поиска минимального элемента квадратной матрицы, номера его строки и столбца. Совет. Можно искать минимальный элемент в полосах строк матрицы в отдельных потоках.
14. Написать параллельную MPI-программу поиска максимального элемента квадратной матрицы. Матрицу разделить на полосы по числу процессов программы. Каждый процесс отыскивает локально максимальный элемент в своей полосе, номер строки и номер столбца. Результаты поиска максимума собираются на нулевом процессе, определяется глобально максимальный элемент, его номер строки и столбца.
15. Написать многопоточную программу, использующую технологию OpenMp, для поиска максимального элемента квадратной матрицы, номера его строки и столбца. Совет. Можно искать максимальный элемент в полосах строк матрицы в отдельных потоках.
16. Написать параллельную MPI-программу, в которой создать коммуникатор в виде квадратной решетки. Затем, процессоры, стоящие на главной диагонали этой решетки должны вывести сообщения о своей работе.

04.06.2014

Тарасов В.Л.