

Задача № 1

(Нахождение интеграла с использованием MPI)

Постановка задачи.

(1)

Решить определенный интеграл (1) методом трапеций.

Предполагается, что запуск исполняемого файла будет происходить с использованием p процессов. Один из p процессов («основной») разбивает отрезок $[0; 1]$ на N малых отрезков длиной Δx (шаг интегрирования), и вычисляет с этим разбиением интеграл в последовательном варианте. Далее этот же процесс разбивает отрезок $[0; 1]$, состоящий из N малых отрезков, на p частей и границы каждой из оставшихся $(p-1)$ частей рассылает остальным $(p-1)$ процессам (с одной из частей отрезка работает сам «основной» процесс). Число N может меняться и задается пользователем.

Каждый из процессов, получивших свои границы части отрезка, должен вычислить свою часть интеграла I_i и отправить ее «основному» процессу.

«Основной» процесс получает все части интеграла от процессов-рабочих и, складывая их, получает исходный интеграл I .

Задание:

1) Вывести на экран в столбик значения частей интеграла I_i , посчитанные каждым из процессов-рабочих с указанием его номера.

2) Вывести на экран значение интеграла I , посчитанное сложением всех частей интеграла, полученных «основным» процессом от процессов-рабочих.

3) Вывести на экран интеграл I_0 , посчитанный «основным» процессом последовательно. Сравнить его со значением I .

4) На одной координатной плоскости построить 3 графика зависимости ускорения S от количества процессов p , где $p = 1, 2, 3, \dots, 8$ для $N = 1000$, $N = 10^6$ и для $N = 10^8$.

Примечания:

1) Подумать над разбивкой отрезка $[0; 1]$ на части, когда N не делится нацело на p .

2) Использовать следующую особенность программирования: если не обращаться ни к какому из процессов посредством конструкции `if`, а написать код в общей части, то этот кусок кода будет выполнен всеми процессами одинаково.

3) Если в общей части программы объявить переменную, то она в разных процессах будет называться одинаково, но может при этом принимать различные значения.