Міністерство освіти і науки України

Одеський національний політехнічний університет

Інститут комп’ютерних систем  
Кафедра комп’ютерних інтелектуальних систем і мереж

Протокол

Лабораторної роботи №4

На тему “ Разработка параллельных программ с использованием интерфейса передачи сообщений MPI. Мини-MPI (старт, финиш, передача и прием сообщений).”

По предмету: «Комп’ютерні інтелектуальні системи»

Виконали:

студенти групи АМ-181

Батаєв М.С.

Карабань О.С.

Борщов М.С.

Перевірив:

Тішин П.М.

Одеса 2021

**Перелік завдань до лабораторної роботи**

1.1. Разработать программу для вычисления значения функции y(x). Вывести на экран результаты и время выполнения программы. Варианты задания функции y(x) приведены в таблице.

1.2. Привести результаты вычислений, время выполнения программ, листинги программ.

**Вариант задания 1**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **№** | **Вид функции** | **Нижний предел a** | **Верхний предел b** | **Шаг дискритезации** |
| 1 |  | 0 | π/2 | 0.00001 |
| 2 | π/2 | π | 0.00002 |
| 3 | π | 3π/2 | 0.00003 |
| 4 | 3π/2 | 2π | 0.00005 |
| 5 | 0 | π | 0.0001 |

Ход работы

Листинг программы:

#include <mpi.h>

#include <conio.h>

#include <time.h>

#include <iostream>

#include <windows.h>

#include <math.h>

const double pi = atan(1) \* 4;

using namespace std;

int main(int argc, char\* argv[])

{

int size, rank, n;

double y, a, b, result, endwtime;

double startwtime = 0.0;

MPI\_Status status;

MPI\_Init(&argc, &argv);

MPI\_Comm\_size(MPI\_COMM\_WORLD, &size); //определяем кол-во процессов

MPI\_Comm\_rank(MPI\_COMM\_WORLD, &rank); //определяем ранк текущего процесса

if (rank == 0)

{

startwtime = MPI\_Wtime();

}

MPI\_Bcast(&n, 1, MPI\_DOUBLE, 0, MPI\_COMM\_WORLD);

y = -0.25 \* cos(0) + 0.25 \* cos(pi);

MPI\_Reduce(&y, &result, 1, MPI\_DOUBLE, MPI\_SUM, 0, MPI\_COMM\_WORLD);

if (rank == 0)

{

cout << "Result: " << result << endl;

endwtime = MPI\_Wtime();

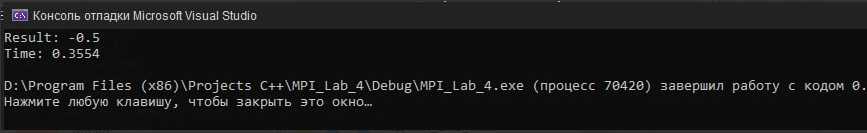
cout << "Time: " << (endwtime - startwtime) \* 1000 << endl;

}

MPI\_Finalize();

return 0;

}

**Результат работы программы:**

**Вывод:**

В этой лабораторной работе мы научились вычислять интегралы при помощи MPI, а также определили время выполнения программы. Были задействованы методы MPI\_Reduce, MPI\_Bcast, MPI\_init и

Finalize().