

实验三 MPI 实现梯形积分

庄清惠 14331401

实验目的： 用 MPI 接口编程实现梯形积分。

代码解析：（完整代码请查看.c 文件）

初始化

```
MPI_Init(&argc, &argv);
int my_rank, comm_sz, n, local_n;
double a, b;
double local = 0.0, total = 0.0;
MPI_Comm_rank(MPI_COMM_WORLD, &my_rank);
MPI_Comm_size(MPI_COMM_WORLD, &comm_sz);
```

my_rank 表示进程 id, comm_sz 表示总进程数, local_n 表示该进程要处理的区间数。local 表示该区间的面积计算值, total 为总面积。

读入数据

```
void Get_input(int my_rank, int comm_sz, double* a_p, double* b_p,
               int* n_p) {
    int dest;
    if (my_rank == 0) {
        printf("Enter a, b and n\n");
        scanf("%lf%lf%d", a_p, b_p, n_p);
        for (dest = 1; dest < comm_sz; ++dest) {
            MPI_Send(a_p, 1, MPI_DOUBLE, dest, 0, MPI_COMM_WORLD);
            MPI_Send(b_p, 1, MPI_DOUBLE, dest, 0, MPI_COMM_WORLD);
            MPI_Send(n_p, 1, MPI_INT, dest, 0, MPI_COMM_WORLD);
        }
    } else {
        MPI_Recv(a_p, 1, MPI_DOUBLE, 0, 0, MPI_COMM_WORLD, MPI_STATUS_IGNORE);
        MPI_Recv(b_p, 1, MPI_DOUBLE, 0, 0, MPI_COMM_WORLD, MPI_STATUS_IGNORE);
        MPI_Recv(n_p, 1, MPI_INT, 0, 0, MPI_COMM_WORLD, MPI_STATUS_IGNORE);
    }
}
```

零号进程为主进程，输出提示“Enter a,b and n” a, b 代表积分区间，n 代表划分的数目。零号进程读取数据并发送给其他进程，其他进程接收数据。

计算积分

```
double func(double x) {
    return x * x;
}
```

func 为用户自定义的被积函数，Trap 用来计算梯形的面积。

直接调用 MPI_Reduce 将计算结果汇总到主进程。

```
MPI_Reduce(&local, &total, 1, MPI_DOUBLE, MPI_SUM, 0,
           MPI_COMM_WORLD);
```

打印结果

```
if (my_rank == 0) {
    printf("With n = %d trapezoids, our estimate\n", n);
    printf("of the integral from %f to %f = %.15e\n",
           a, b, total);
}
```

主进程打印计算结果，并显示相关信息。

结束程序

```
MPI_Finalize();
return 0;
```

运行结果：

```
zhuangqh@ubuntu:~/Cpp_homework/CUDA/experiment3$ mpirun -n 10 ./jifen
Enter a, b and n
0 1 100
With n = 100 trapezoids, our estimate
of the integral from 0.000000 to 1.000000 = 3.333500000000000e-01
zhuangqh@ubuntu:~/Cpp_homework/CUDA/experiment3$
```

运行结果正确