

MPI作业：蒙特卡洛方法估计 π 值

蒙特卡洛方法介绍

[蒙特卡洛方法](#)实现计算圆周率的方法比较简单，其思想是假设我们向一个正方形的标靶上随机投掷飞镖，靶心在正中央，标靶的长和宽都是2英尺。同时假设有一个圆与标靶内切。圆的半径是1英尺，面积是 π 平方英尺。如果击中点在标靶上是均匀分布的（我们总会击中正方形），那么飞镖击中圆的数量近似满足等式

$$\frac{\text{击中圆内的投掷次数}}{\text{全部的投掷次数}} = \frac{\pi}{4}$$

因为圆包含的面积与正方形面积的比值是 $\pi/4$ 。

我们可以用这个公式和随机数产生器来估计 π 的值。伪代码如下：

```
number_in_circle=0;
for(toss=0;toss<number_of_tosses;toss++){
    x=random double between -1 and 1;
    y=random double between -1 and 1;
    distance_squared=x*x+y*y;
    if(distance_squared<=1) number_in_circle++;
}
pi_estimate=4*number_in_circle/((double)number_in_tosses);
```

作业及要求

编写一个用上述方法估计 π 的MPI程序，**进程0读入总的投掷次数**，并把他们广播(MPI_Bcast)给各个进程，使用**MPI_Reduce**求出局部变量**number_in_cycle**的全局总和，并让0进程输出估计的 π 值

```
printf("pi estimate = %f\n", 估计的pi值);
```

为了获得较精确的 π 估计值，击中圆内部的次数和投掷次数可能要使用**Long Long int**类型的数值来表示。

要求：

1. 使用MPI库完成作业
2. 程序可以运行，并在进程数为8，投掷次数为 10^8 次方以上时，能够精确到小数点后两位。即
`mpiexec -np 8 二进制文件`后输出的 π 结果3.14xxxxxx。

```
$ mpiexec -n 8 ./a.out
Enter the total number of tosses
100000000
pi estimate = 3.141404
```

3. 提交方式，源代码C或Cpp文件，命名为“学号_姓名_MPI”，提交到发送到群里面的问卷星上。截止日期3.30号24:00点前。可重复提交。