

Find_bins 函数解析:

```
/*-----*/
void Find_bins(
    int bin_counts[]      /* out */,
    float local_data[]    /* in */,
    int loc_bin_cts[]     /* out */,
    int local_data_count /* in */,
    float bin_maxes[]     /* in */,
    int bin_count         /* in */,
    float min_meas       /* in */,
    MPI_Comm comm){

    int i,bin;
    for (i = 0; i < local_data_count; i++){
        bin = Which_bin(local_data[i], bin_maxes, bin_count, min_meas);
        loc_bin_cts[bin]++;
    }

    /*****
    /
    MPI_Reduce(loc_bin_cts, bin_counts, bin_count, MPI_INT, MPI_SUM, 0,
comm);
    *****/
} /* Find_bins */
```

MPI_Reduce 函数参数介绍:

```
int MPI_Reduce(
void *input_data, /*指向发送消息的内存块的指针 */
void *output_data, /*指向接收（输出）消息的内存块的指针 */
int count, /*数据量*/
MPI_Datatype datatype, /*数据类型*/
MPI_Op operator, /*规约操作*/
int dest, /*要接收（输出）消息的进程的进程号*/
MPI_Comm comm); /*通信器，指定通信范围*/
```

所以 MPI_Reduce 函数的作用为：将所有的发送信息进行同一个操作。

在本题中的作用为：将每个进程中的 loc_bin_cts 中的每个同一位置的元素进行整体求和 (MPI_SUM)，并将其输出消息发送至 bin_counts，且最后将 bin_counts 中的消息发送至进程 0。

Which_bin 函数解析:

```
/*-----*/
int Which_bin(float data, float bin_maxes[], int bin_count,
              float min_meas) {

    /******
    /
    int down=0;//bin 标号的下界
    int up=bin_count-1;//bin 标号的上界
    int mid;
    float bin_max,bin_min;//用来记录某一个 bin 的上下界

    while(down<=up){//开始进行二分查找
        mid=(up+down)/2;
        bin_max=bin_maxes[mid];//当前查找的 bin 的上界
        if(mid==0){//特殊情况，若当前 bin 为 0 号 bin
            bin_min=min_meas;
        }
        else{//通常情况下
            bin_min=bin_maxes[mid-1];//当前查找的 bin 的下界
        }
        if(data>=bin_max){
            down=mid+1;//若数据大于当前 bin[mid]的最大值，则 down=mid+1
        }
        else if(data<bin_min){
            up=mid-1;//若数据小于当前 bin[mid]的最小值，则 up=mid-1;
        }
        else{
            return mid;//若数据在 bin[mid]的上下界范围之内，则找到，返回当前
            bin 的标号，即 mid
        }
    }

    /******

    printf("Uh oh . . .\n");//二分查找找不到则打印该信息
    return 0;
} /* Which_bin */
```

该函数中的补充部分主要是对数据 data 所属于的 bin 进行二分查找以归类，解析可见代码注释。

结果展示（在超算习堂的自由编程平台上编译运行）：

输入为：

Enter the number of bins: 10

Enter the minimum measurement: 0

Enter the maximum measurement: 5

Enter the number of data: 100

运行结果

```
===== ERROR =====

===== OUTPUT =====
Enter the number of bins
Enter the minimum measurement
Enter the maximum measurement
Enter the number of data
0.000-0.500: XXXXXXXX
0.500-1.000: XXXXXXXXXXXX
1.000-1.500: XXXXXXXXXXXX
1.500-2.000: XXXXXXXXXXXX
2.000-2.500: XXXXXXXXXXXX
2.500-3.000: XXXXXXXXXXXX
3.000-3.500: XXX
3.500-4.000: XXXXXXXX
4.000-4.500: XXXXXXXX
4.500-5.000: XXXXXXXXXXXXXXXX

===== REPORT =====
```