分散情報システム第 8 回課題

ソースコード

```
1  #include <stdio.h>
2  #include "mpi.h"
3
4  #define SIZE 8
5
6  int main(int argc, char **argv)
7  {
8          int my_rank;
9          int np;
10         int i;
11         int sbuf_a[SIZE];
12         int sbuf_b[SIZE];
13         int rbuf_a[SIZE];
14         int rbuf_b[SIZE];
15         int sum;
16         int r;
17         MPI_Status status;
18
19         MPI_Init(&argc, &argv);
20
21         MPI_Comm_rank(MPI_COMM_WORLD, &my_rank);
22
23         MPI_Comm_size(MPI_COMM_WORLD, &np);
24
25         if (my_rank == 0) {
26                 for (i = 0; i < SIZE; i++) {
27                         sbuf_a[i] = i;
28                         sbuf_b[i] = i;
29                 }
30         }
31
32         MPI_Scatter(sbuf_a, SIZE/np, MPI_INT, rbuf_a, SIZE/np, MPI_INT, 0,
               MPI_COMM_WORLD);
33         MPI_Scatter(sbuf_b, SIZE/np, MPI_INT, rbuf_b, SIZE/np, MPI_INT, 0,
               MPI_COMM_WORLD);
34         for (i = 0; i < SIZE/np; i++) {
35                 sum += rbuf_a[i] * rbuf_b[i];
36         }
37         printf("rank %d:sum=%d\n", my_rank, sum);
38
39         MPI_Reduce(&sum, &r, 1, MPI_INT, MPI_SUM, 0, MPI_COMM_WORLD);
40         if (my_rank == 0) {
41                 printf("inner product=%d\n", r);
```

```
42          }
43
44          MPI_Finalize();
45          return 0;
46  }
```