

```
1  /* spedizione della terza colonna di una matrice dal processo 0 al processo 1
2  *      process 1
3  *
4  * Note: Dovrebbe eseguito su due processi.
5  *
6  */
7  #include <stdio.h>
8  #include "mpi.h"
9
10 main(int argc, char* argv[]) {
11     int p;
12     int my_rank;
13     float A[10][10];
14     MPI_Status status;
15     MPI_Datatype column_mpi_t;
16     int i, j;
17
18     MPI_Init(&argc, &argv);
19     MPI_Comm_rank(MPI_COMM_WORLD, &my_rank);
20
21     MPI_Type_vector(10, 1, 10, MPI_FLOAT, &column_mpi_t);
22     MPI_Type_commit(&column_mpi_t);
23
24     if (my_rank == 0) {
25         for (i = 0; i < 10; i++)
26             for (j = 0; j < 10; j++)
27                 A[i][j] = (float) j;
28         MPI_Send(&A[0][2], 1, column_mpi_t, 1, 0,
29                 MPI_COMM_WORLD);
30     } else { /* my_rank = 1 */
31         MPI_Recv(&A[0][2], 1, column_mpi_t, 0, 0,
32                 MPI_COMM_WORLD, &status);
33         for (i = 0; i < 10; i++)
34             printf("%3.1f ", A[i][2]);
35         printf("\n");
36     }
37
38     MPI_Finalize();
39 } /* main */
40
```