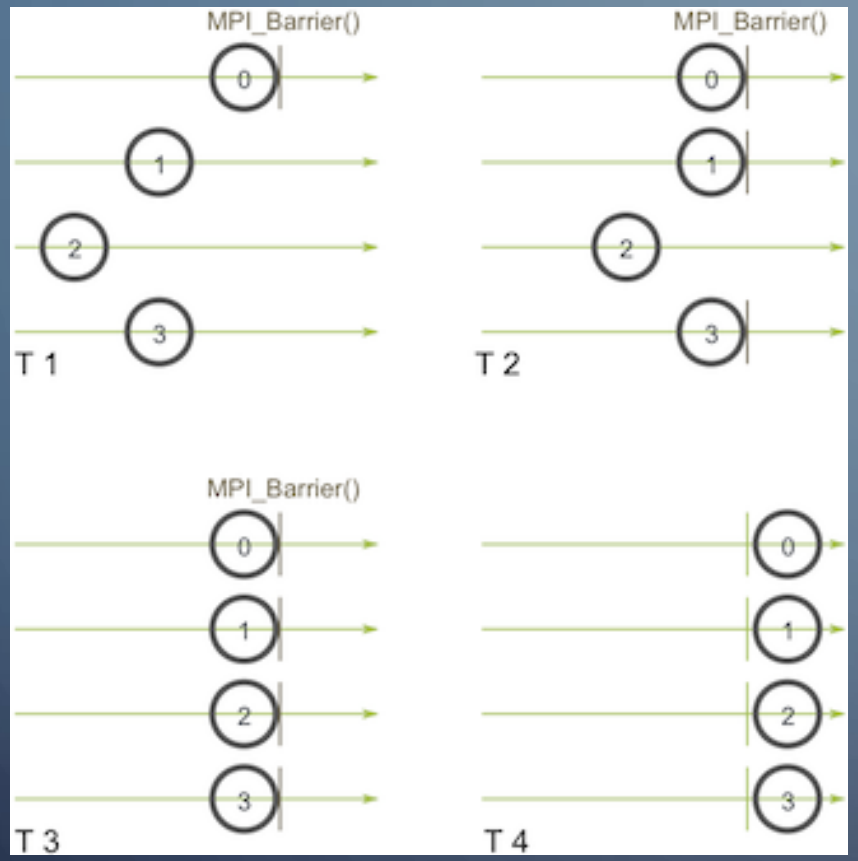
BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ

BLM541 PARALEL VE DAĞITIK PROGRAMLAMA

LECTURE 3: MPI'DA TOPLU HABERLEŞME

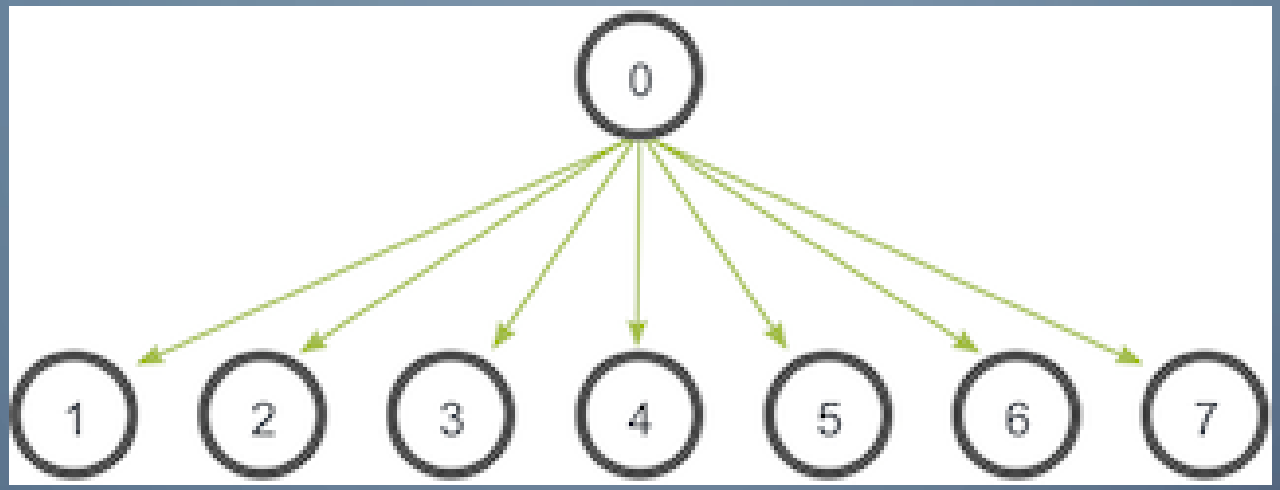
DR. ÖĞR. ÜYESİ SÜHA TUNA

MPI_BARRIER



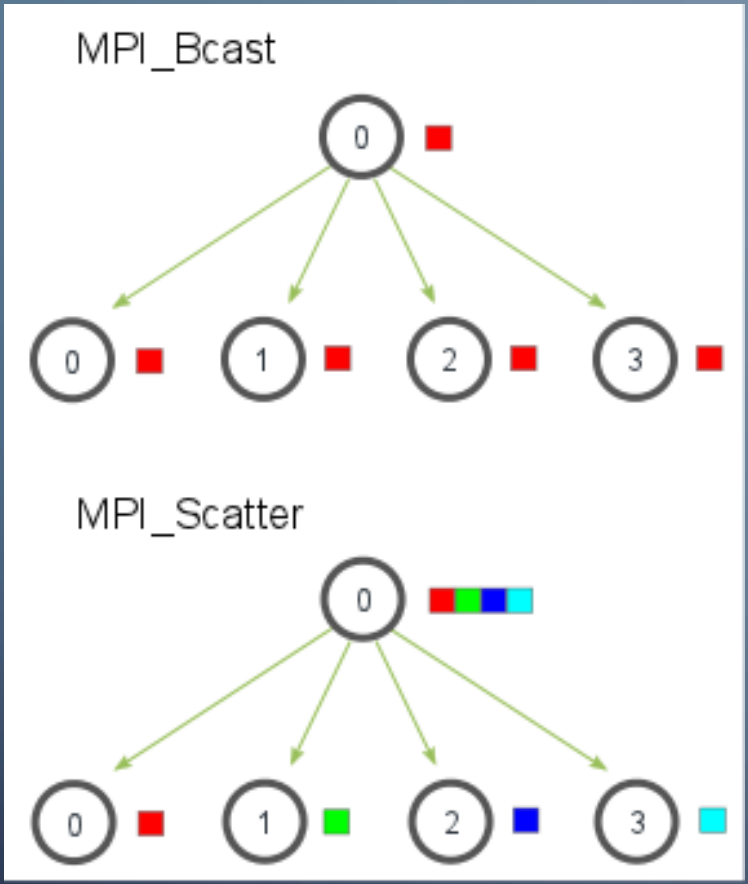

```
MPI_Barrier(MPI_Comm communicator)
```

MPI_BROADCAST



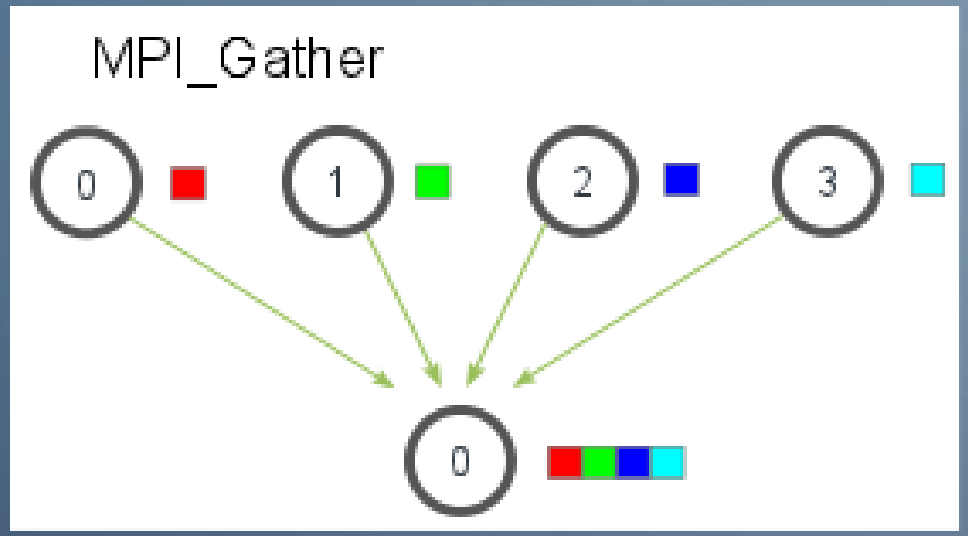

```
MPI_Bcast(
    void* data,
    int count,
    MPI_Datatype datatype,
    int root,
    MPI_Comm communicator)
```

MPI_SCATTER



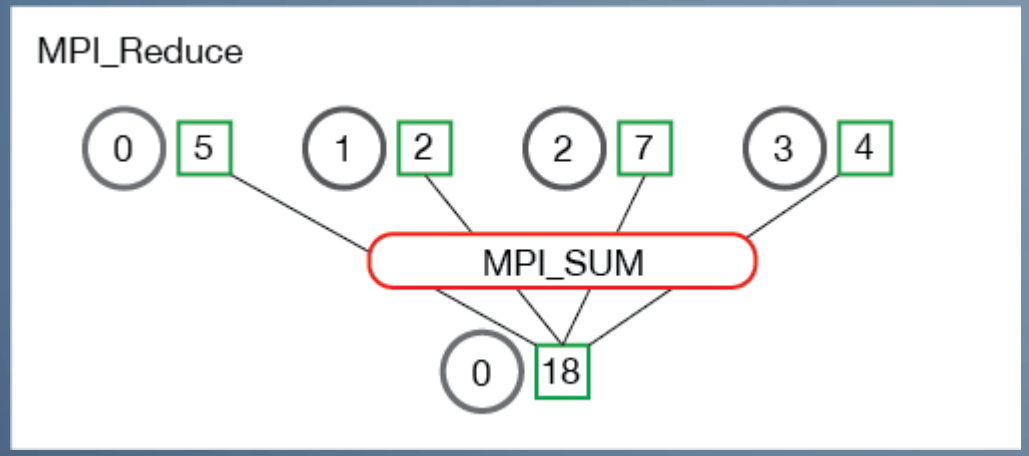

```
MPI_Scatter(
    void* send_data,
    int send_count,
    MPI_Datatype send_datatype,
    void* recv_data,
    int recv_count,
    MPI_Datatype recv_datatype,
    int root,
    MPI_Comm communicator)
```

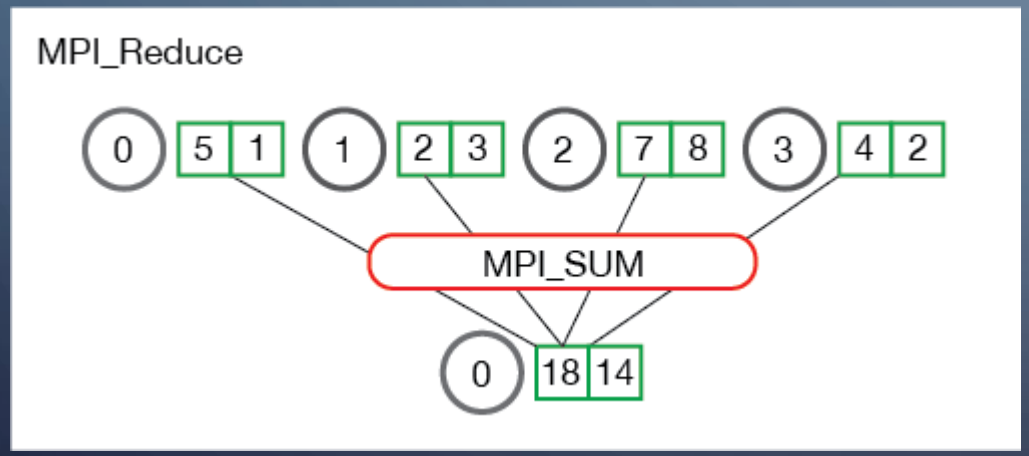
MPI_GATHER




```
MPI_Gather(
    void* send_data,
    int send_count,
    MPI_Datatype send_datatype,
    void* recv_data,
    int recv_count,
    MPI_Datatype recv_datatype,
    int root,
    MPI_Comm communicator)
```

MPI_REDUCE






```
MPI_Reduce(
    void* send_data,
    void* recv_data,
    int count,
    MPI_Datatype datatype,
    MPI_Op op,
    int root,
    MPI_Comm communicator)
```

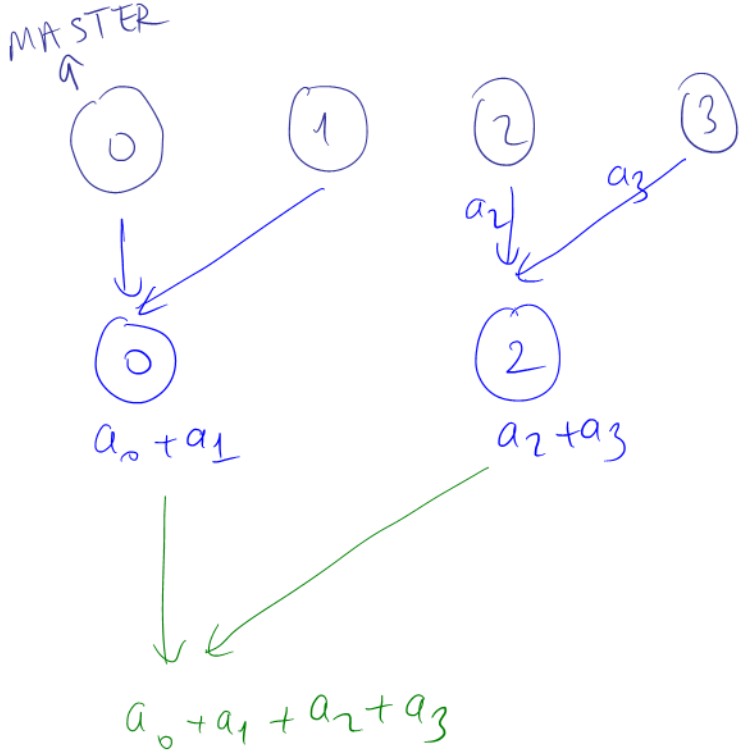
MPI_MAX - Returns the maximum element.
MPI_MIN - Returns the minimum element.
MPI_SUM - Sums the elements.
MPI_PROD - Multiplies all elements.
MPI_LAND - Performs a logical and across the elements.
MPI_LOR - Performs a logical or across the elements.
MPI_BAND - Performs a bitwise and across the bits of the elements.
MPI_BOR - Performs a bitwise or across the bits of the elements.

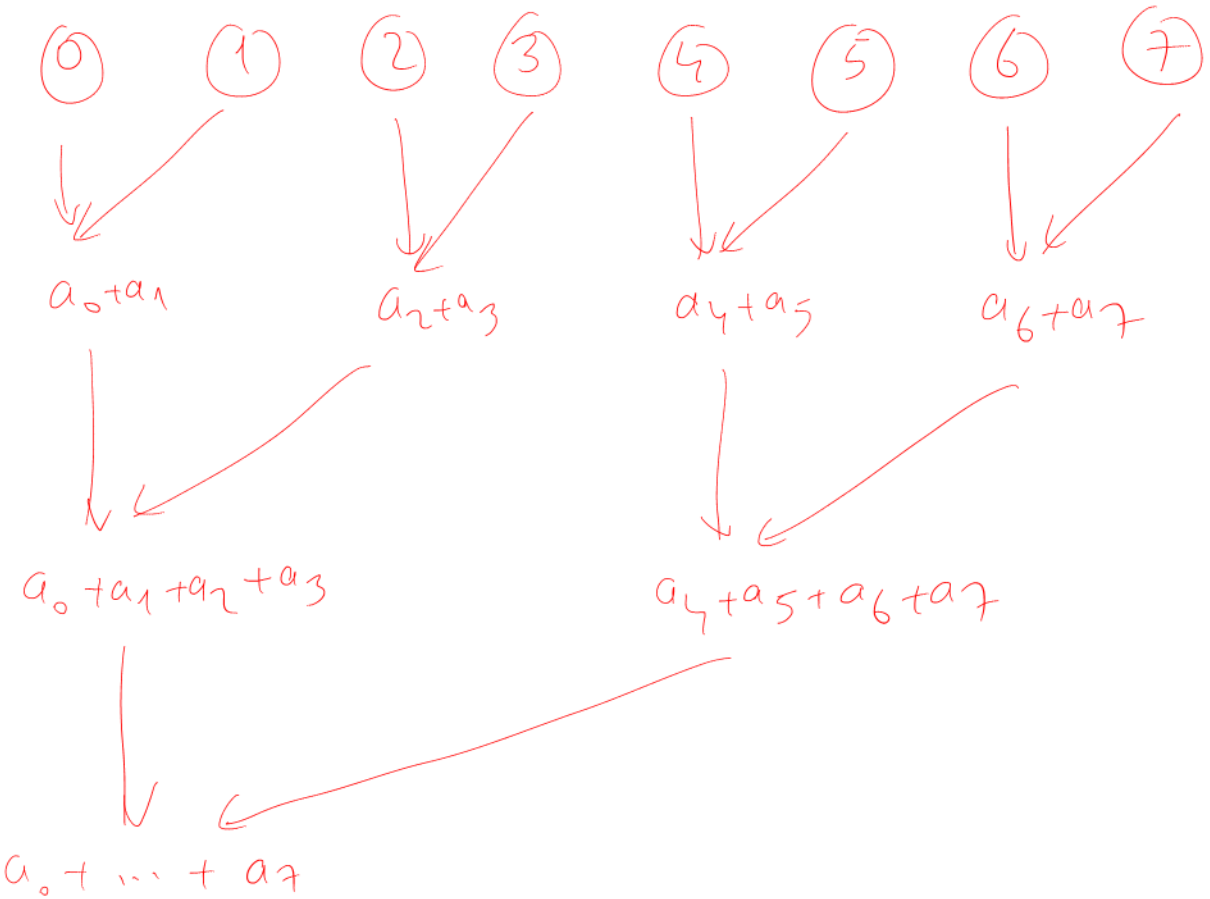
Homework II :

Write your own MPI-Reduce(...);

Write a generic function named myReduce which simulates MPI-Reduce subroutine for MPI-SUM operation. (using MPI-Send and MPI-Recv routines)

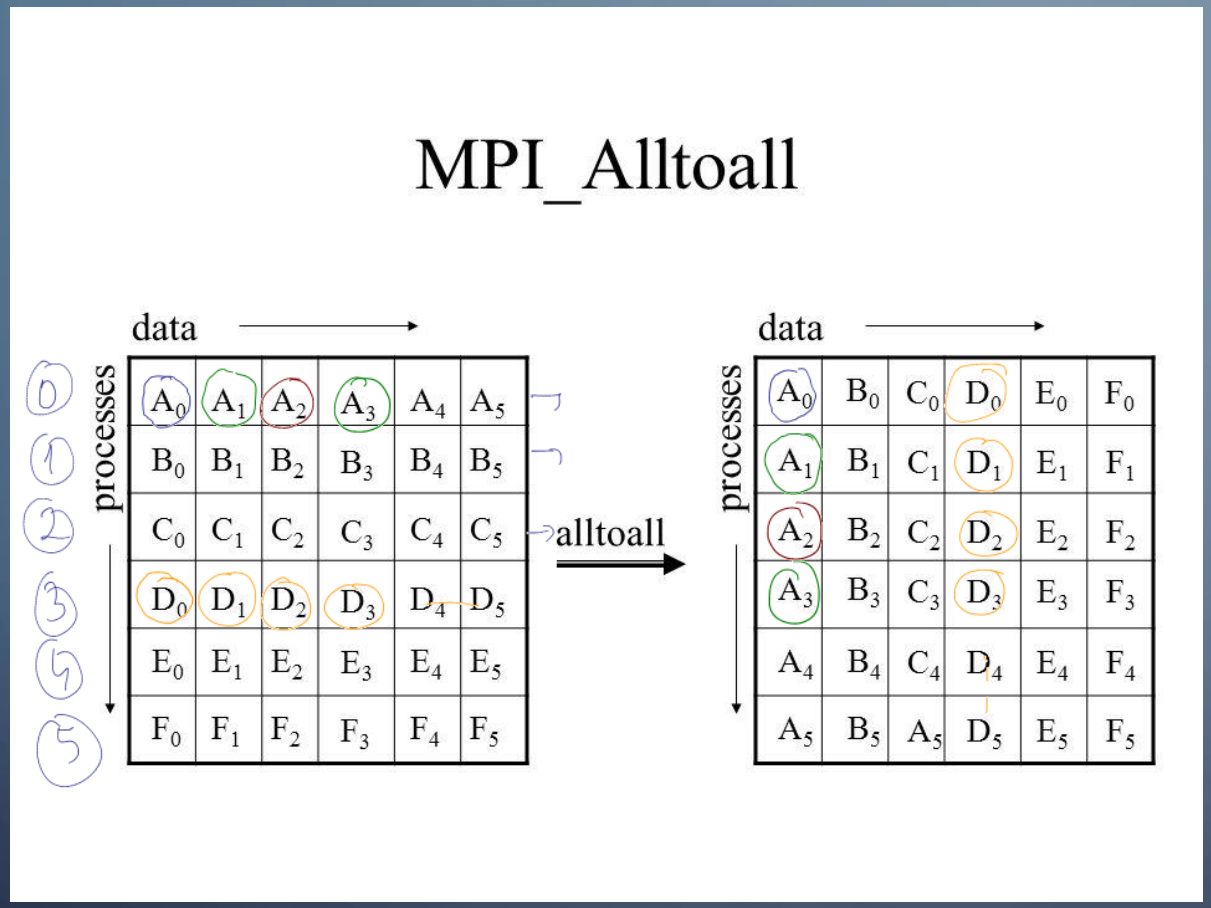
- Process number should be selected as 2^n.





- Compare your myReduce and MPI-Reduce in sense of Wall Clock Time.

MPI_ALLTOALL




```
int MPI_Alltoall(
        const void *sendbuf,
        int sendcount,
        MPI_Datatype sendtype,
        void *recvbuf,
        int recvcount,
        MPI_Datatype recvtype,
        MPI_Comm comm)
```

0. process : MPI_Scatter
1. " " "
2. " " "

MPI_SCATTERV

Vector

```
int MPI_Scatterv(const void *sendbuf,
    const int *sendcounts,
    const int *displs,
    MPI_Datatype sendtype,
    void *recvbuf,
    int recvcount,
    MPI_Datatype recvtype,
    int root,
    MPI_Comm comm)
```

displacements

MPI_Scatter → Equal size

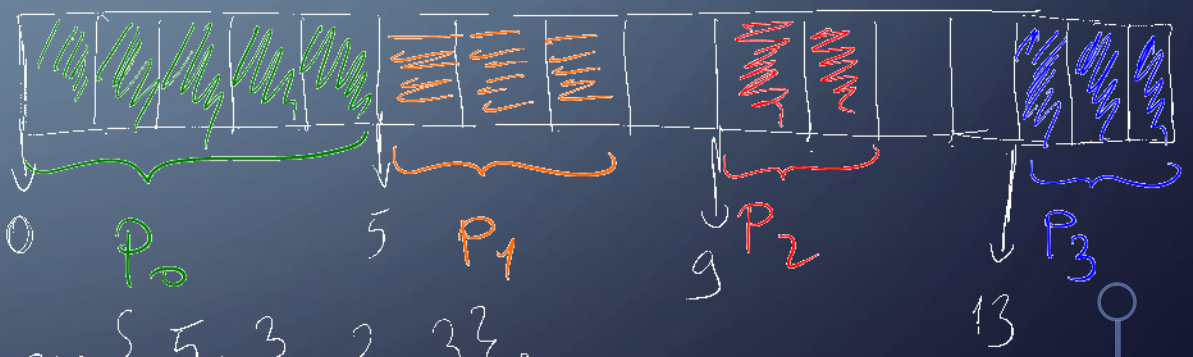
MPI_ScatterV → Vectors not needed to be equal.

sendcounts [rank]

Who scatters?

MASTER: 4 proc

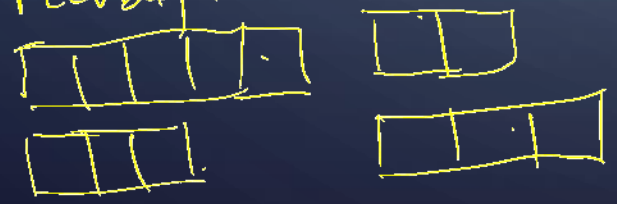
sendbuf



sendcounts = {5, 3, 2, 3};

displs = {0, 5, 9, 13};

recvbuf:



MPI_GATHERV

```
int MPI_Gatherv(
      const void *sendbuf,
      int sendcount,
      MPI_Datatype sendtype,
      void *recvbuf,
      const int *recvcounts,
      const int *displs,
      MPI_Datatype recvtype,
      int root,
      MPI_Comm comm)
```