# Blocking Operations

# Blocking Operations

- The function MPI_Recv can be started whether or not a matching send has been posted.
- It is a **blocking** operation.
- The function MPI_Send can be started whether or not a matching receive has been posted.
- Thus, it is also a **blocking** operation.

# Blocking Operations

- In ill-constructed programs, **blocking** operations may lead to **deadlock**, where all processes are blocked and no progress occurs.

# Example of Deadlock

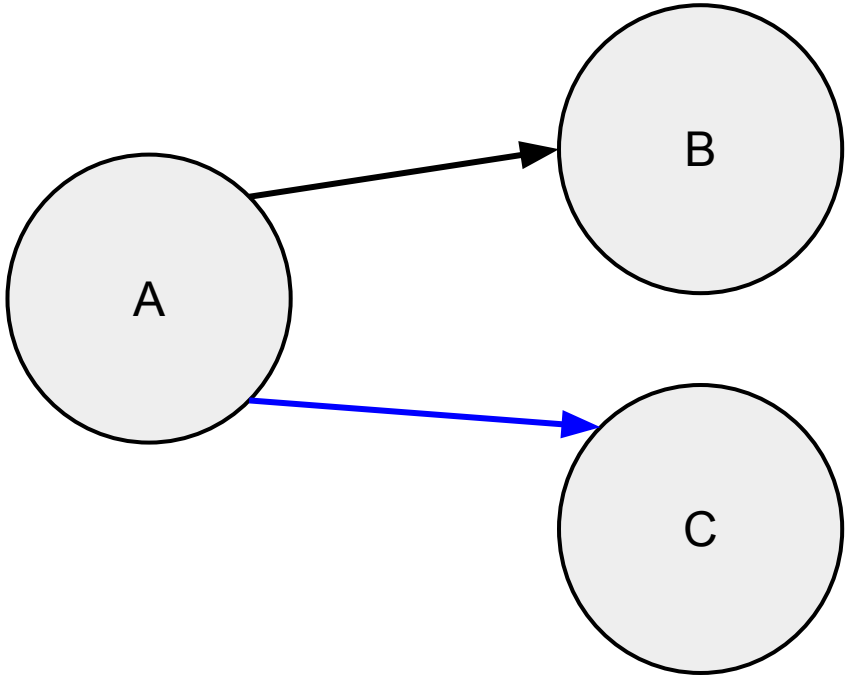
```
MPI_Comm_rank(MPI_COMM_WORLD,&rank);
if (rank == 0) {
MPI_Recv(msg,count,MPI_REAL,1,tag,MPI_COMM_WORLD,&status);
MPI_Send(msg,count,MPI_REAL,1,tag,MPI_COMM_WORLD);
}

else if (rank == 1) {
MPI_Recv(msg,count,MPI_REAL,0,tag,MPI_COMM_WORLD,&status);
MPI_Send(msg,count,MPI_REAL,0,tag,MPI_COMM_WORLD);
}
```

# Messages Order

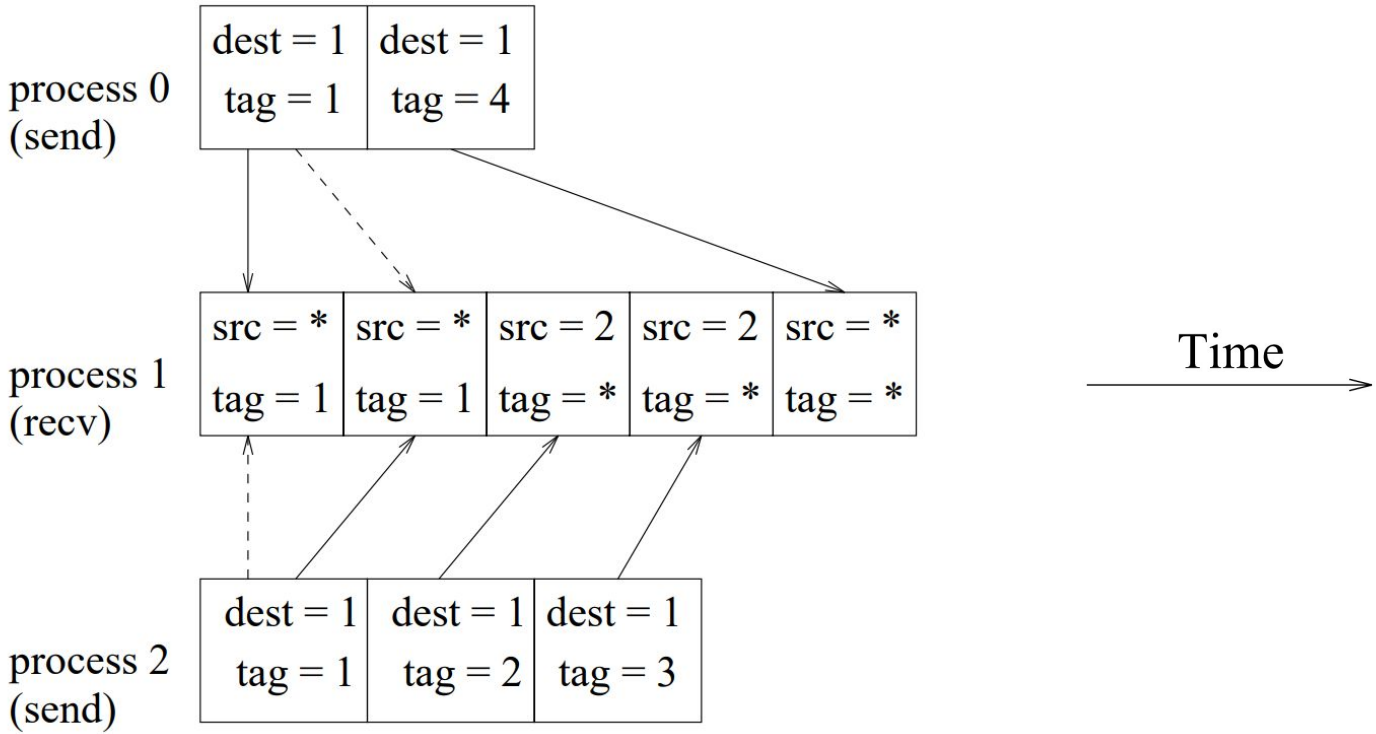
- Messages are non-overtaking.

# Messages Order

- Messages are non-overtaking.



# Messages Order



# Messages Order

- If a sender sends two messages in succession to the same destination and both match the same receive then the receive cannot get the second message if the first message is still pending.

- If a receiver posts two receives in succession and both match the same message then the second receive operation cannot be satisfied by this message if the first receive is still pending.