

1. ROZSYŁANIE/ZBIERANIE DANYCH

MPI_Bcast - Funkcja rozsyła komunikat do wszystkich procesów w obrębie komunikatora comm poczynając od procesu root. Pozostałe argumenty - podobnie jak w MPI_Send().

```
int MPI_Bcast( void * buffer, int count, MPI_Datatype datatype, int root, MPI_Comm comm );
```

buffer	adres do bufora z danymi do wysłania/odebrania.
count	ilość elementów w buforze.
datatype	typ danych.
root	numer procesu rozsyłającego dane.
comm	komunikator.

MPI_Reduce - Obliczanie równoległe. Bardzo ważna funkcja - pozwala wykonać na przykład sumowanie wszystkich częściowych wyników otrzymanych w procesach i umieszczenie wyniku w zmiennej. Argument root wskazuje dla którego procesu wynik ma być umieszczony w zmiennej result. Oto przykład użycia tej funkcji:

```
MPI_Reduce(&suma_cz, &suma, 1, MPI_FLOAT, MPI_SUM, 0, MPI_COMM_WORLD);
```

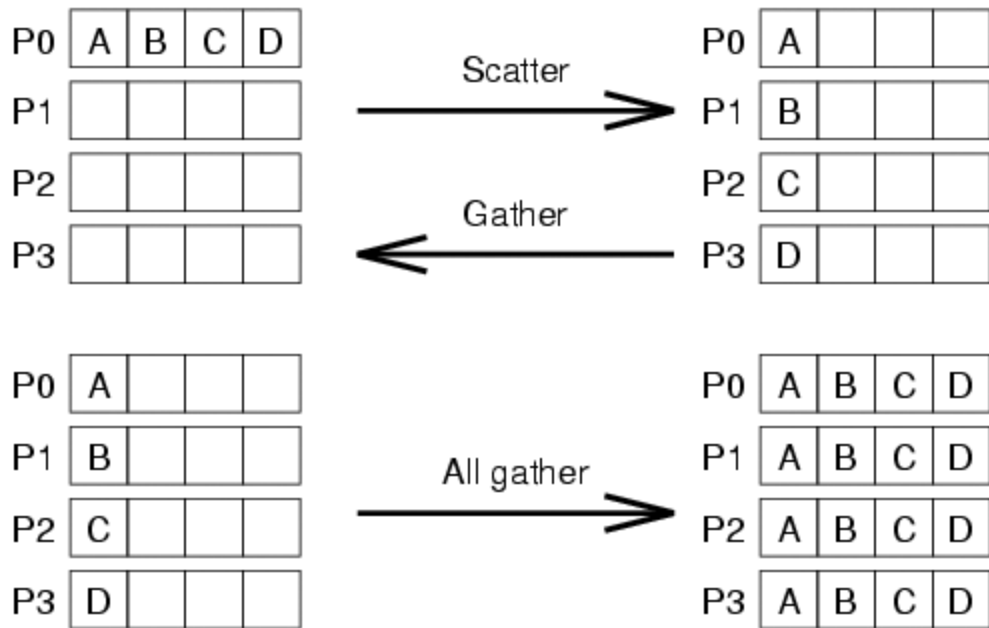
Przykładowe operatory to MPI_MAX, MPI_MIN, MPI_SUM - kompletna lista znajduje się w dokumentacji. Istnieje również możliwość definiowania własnych operatorów dla funkcji MPI_Reduce().

```
int MPI_Reduce( void * sendbuf, void * recvbuf, int count, MPI_Datatype datatype, MPI_Op op, int root, MPI_Comm comm );
```

sendbuff	adres do bufora z danymi.
recvbuff	adres na bufor w którym mają znaleźć się wyniki obliczeń.
count	ilość elementów w buforze.
datatype	typ danych.
op	działanie do wykonania.
root	numer procesu rozsyłającego dane.
comm	komunikator.

MPI_Gather - zbieranie wszyscy do jednego,

MPI_Scatter - rozpraszanie jeden do wszystkich,



INSTALACJA ŚRODOWISKA MPI (UBUNTU):

```
sudo apt-get install libcr-dev mpich mpich-doc
```

KOMPILOWANIE PROGRAMU W ŚRODOWISKU MPI:

```
mpic++ program.cpp -o program
```

URUCHOMIENIE PROGRAMU:

```
mpirun -np 4 ./program
```

