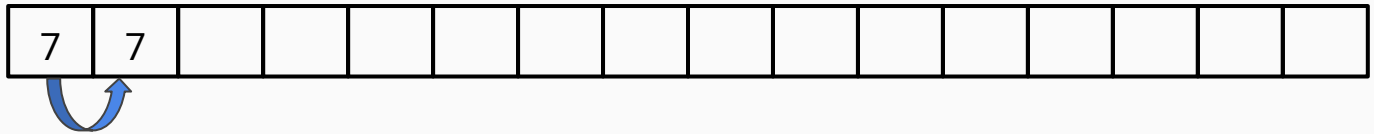
Broadcast

În acest exemplu, rulăm cu 16 procese (fiecare proces, mai puțin procesul 0, nu are nicio valoare)

```
for (pas = 1; pas < nr_procese; pas *= 2)
    if (rank < pas and rank + pas < nr_procese)
        trimite la procesul cu rank-ul [rank + pas]
    else if (rank >= pas and rank < pas * 2)
        primește de la procesul cu rank-ul [rank - pas]
```

Broadcast

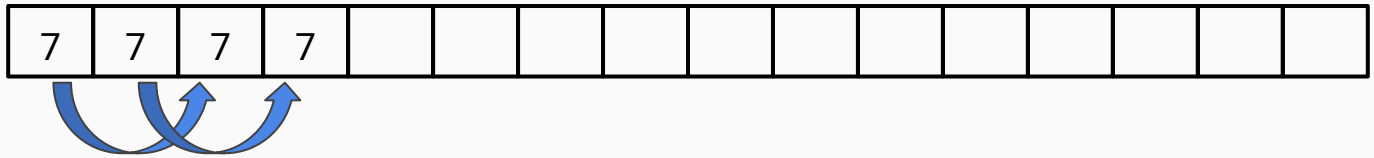
pas = 1



- procesul 0 (0 < 1, 0 + 1 < 16) trimite procesului 1 (0 + 1, 1 >= 1, 1 < 2 * 1)

Broadcast

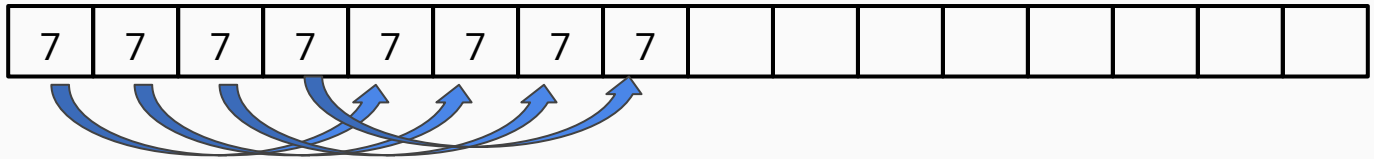
pas = 2



- procesul 0 (0 < 2, 0 + 2 < 16) trimite procesului 2 (0 + 2, 2 >= 2, 2 < 2 * 2)
- procesul 1 (1 < 2, 1 + 2 < 16) trimite procesului 3 (1 + 2, 3 >= 2, 3 < 2 * 2)

Broadcast

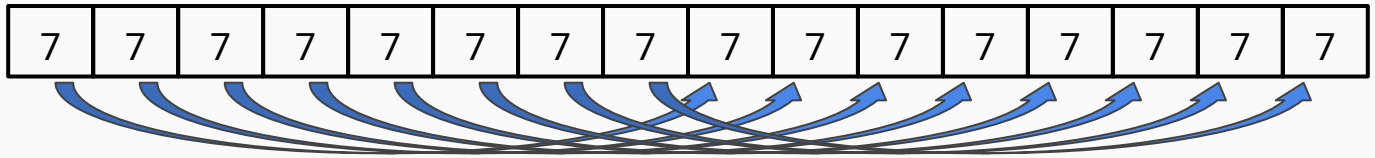
pas = 4



- procesul 0 (0 < 4, 0 + 4 < 16) trimite procesului 4 (0 + 4, 4 >= 4, 4 < 2 * 4)
- procesul 1 (1 < 4, 1 + 4 < 16) trimite procesului 5 (1 + 4, 5 >= 4, 5 < 2 * 4)
- procesul 2 (2 < 4, 2 + 4 < 16) trimite procesului 6 (2 + 4, 6 >= 4, 6 < 2 * 4)
- procesul 3 (3 < 4, 3 + 4 < 16) trimite procesului 7 (3 + 4, 7 >= 4, 7 < 2 * 4)

Broadcast

pas = 8



- procesul 0 (0 < 8, 0 + 8 < 16) trimite procesului 8 (0 + 8, 8 >= 8, 8 < 2 * 8)
- procesul 1 (1 < 8, 1 + 8 < 16) trimite procesului 9 (1 + 8, 9 >= 8, 9 < 2 * 8)
- procesul 2 (2 < 8, 2 + 8 < 16) trimite procesului 10 (2 + 8, 10 >= 8, 10 < 2 * 8)
- procesul 3 (3 < 8, 3 + 8 < 16) trimite procesului 11 (3 + 8, 11 >= 8, 11 < 2 * 8)
- procesul 4 (4 < 8, 4 + 8 < 16) trimite procesului 12 (4 + 8, 12 >= 8, 12 < 2 * 8)
- procesul 5 (5 < 8, 5 + 8 < 16) trimite procesului 13 (5 + 8, 13 >= 8, 13 < 2 * 8)
- procesul 6 (6 < 8, 6 + 8 < 16) trimite procesului 14 (6 + 8, 14 >= 8, 14 < 2 * 8)
- procesul 7 (7 < 8, 7 + 8 < 16) trimite procesului 15 (7 + 8, 15 >= 8, 15 < 2 * 8)