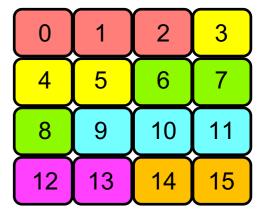
## "Consecutive Slices" Parallel Pizza-Eating Algorithm (does <u>not</u> assume N is evenly divisible by P)

- 1. Define N: How many slices of pizza?
- 2. Define P: How many people?
- Compute slicesPP1 = ceiling(float(N)/P).
- 4. Compute slicesPP2 = slicesPP1 1.
- 6. Assign each slice of pizza a unique number (0..N-1).

5. Compute remnants = N % P (the remainder of N/P).

- o. Assign each since of pizza a unique number (0...14-1).
- 7. Get your own personal, unique id number (0..P-1).
- 8. Compute start = id \* slicesPP1; stop = start + slicesPP1.
  9. If (remnants > 0) AND (id >= remnants), recompute: start = remnants \* slicesPP1 + (id - remnants) \* slicesPP2; stop = start + slicesPP2.
- 10. For (s = start; s < stop; ++s): Eat slice s.



remnants: 4

id:

start:

stop:

N:

**P**:

slicesPP1:

slicesPP2:

16

6

3

0

3

12

14 16

5

14

s (slice #s eaten): 0-2 3-5 6-8 9-11 12-13 14