- How will your team communicate (slack, etc.)
- Algorithms you will implement, including pseudo-code
- What versions do you plan to compare

How we will communicate:

We will use discord to communicate

The algorithms we will implement are:

Parallel Bitonic Sort

Parallel Algorithms Sorting Algorithms

Parallel Bitonic Sort on a Hypercube

- 1. **procedure** BITONIC SORT(*label*, *d*)
- 2. begin
- 3. for i := 0 to d - 1 do
- for j := i downto 0 do 4.
- 5. **if** (i + 1)st bit of label = j th bit of label **then**
- 6. comp exchange max(j);
- 7. else
- 8. comp exchange min(j);
- 9. end BITONIC SORT

Olga Pearce (TAMU)

Design of Parallel Algorithms

October 2, 2023

Parallel Odd-Even Transposition Sort

Parallel Odd-Even Transposition Sort

```
procedure ODD-EVEN PAR(n)
1.
2.
    begin
3.
       id := process's label
4.
       for i := 1 to n do
5.
       begin
6.
          if i is odd then
7.
              if id is odd then
8.
                 compare-exchange min(id + 1);
9.
              else
10.
                 compare-exchange max(id - 1);
11.
          if i is even then
12.
             if id is even then
                 compare-exchange min(id + 1);
13.
14.
              else
15.
                 compare-exchange max(id - 1);
16.
       end for
17. end ODD-EVEN PAR
```

-Parralel Floyd-Warshall's Algorithm

Parallel Floyd's Algorithm

- 1. **procedure** FLOYD ALL PAIRS PARALLEL (A)
- 2. begin
- 3. $D^{(0)} = A$;
- 4. **for** k := 1 **to** n **do**
- 5. **forall** $P_{i,j}$, where $i, j \le n$, **do in parallel**
- 6. $d^{(k)}_{i,j} := \min d^{(k-1)}_{i,j}, d^{(k-1)}_{i,k} + d^{(k-1)}_{k,j};$
- 7. end FLOYD ALL PAIRS PARALLEL

We will be implementing the parallel versions of these algorithms and making comparisons using MPI and CUDA.