Summing 100,000 Numbers on 100 Processors

Start by distributing 1000 elements of vector A to each of the local memories (in Al) and summing each subset in parallel

```
sum = 0;
for (i = 0; i < 1000; i = i + 1)
     sum = sum + Al[i];       /* sum local array subset */
```

The processors then coordinate in adding together the partial sums (Pn is the number of the processor, send(x,y) sends value y to processor x, and receive() receives a value)

```
half = 100;
limit = 100;
repeat
   half = (half + 1) / 2;     /* dividing line */
   if (Pn >= half && Pn < limit)
      send(Pn-half, sum);
   if (Pn < (limit/2))
      sum = sum + receive();
   limit = half;
until (half == 1);       /* final sum in P0's sum */
```

Team #:_____ 11/30/06
Absent:

2. Trace the second segment of code that adds together the partial sums assuming 10 processors.

sum	sum	sum	sum	sum	sum	sum	sum	sum	sum
P0	P1	P2	P3	P4	P5	P6	P7	P8	P9

Team #:_____ 11/30/06
Absent:

3. For a 64 processor system, compare the interconnection network for each of the following topologies.

	Bus	Ring	Torus	6-cube	Fully connected
Network bandwidth	1				
Bisection bandwidth	1				
Total # of Switches	1				
Links per switch					
Total # of links	1				