1. CSE 6220: HW6

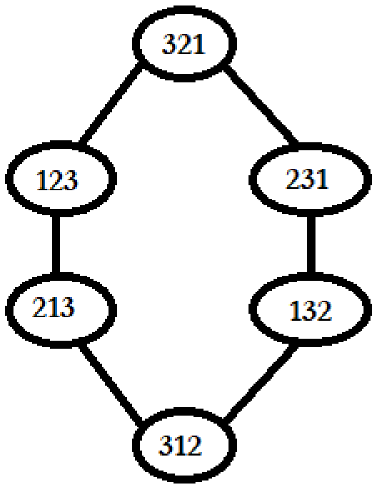
Note: This assignment is submitted by: **Somdut Roy (GTID: sroy86)**

1. Let B be the array as a result of parallel prefix (sum) implemented on array A of size n. The i^{th} value in B is given by the sum of first i values in A. To perform this operation in $O(1)$ time, i processors are used to calculate $B[i]$ for CRCW PRAM sum model. Hence total processors needed to solve this problem$=\sum_{i=1}^{n} i = \frac{n(n+1)}{2} = \Theta(n^2)$.

   Issues with this algorithm: This algorithm has an efficiency of $\Theta(\frac{n}{1.n^2}) = \Theta(\frac{1}{n})$ which implies a decrease in efficiency with increasing problem size. Also availability of $\Theta(n^2)$ processors for a work of size n is not a practical demand to make.

2. To see which topology is better we need to look at (a) the maximum distance between any two nodes, (b) Number of neighbors of each node, (c) bisector width and (d) number of links. In both cases, for a 64-processor configuration we get same figures for all of them i.e. (a) $log(p) = log(64) = 6$, (b) $log(p) = log(64) = 6$, (c) $\frac{p}{2} = \frac{64}{2} = 32$ and (d) $\frac{plog(p)}{2} = \frac{64log(64)}{2} = 192$. Hence we can't conclude on the superiority of any of the two choices.

3. (a) S_3:



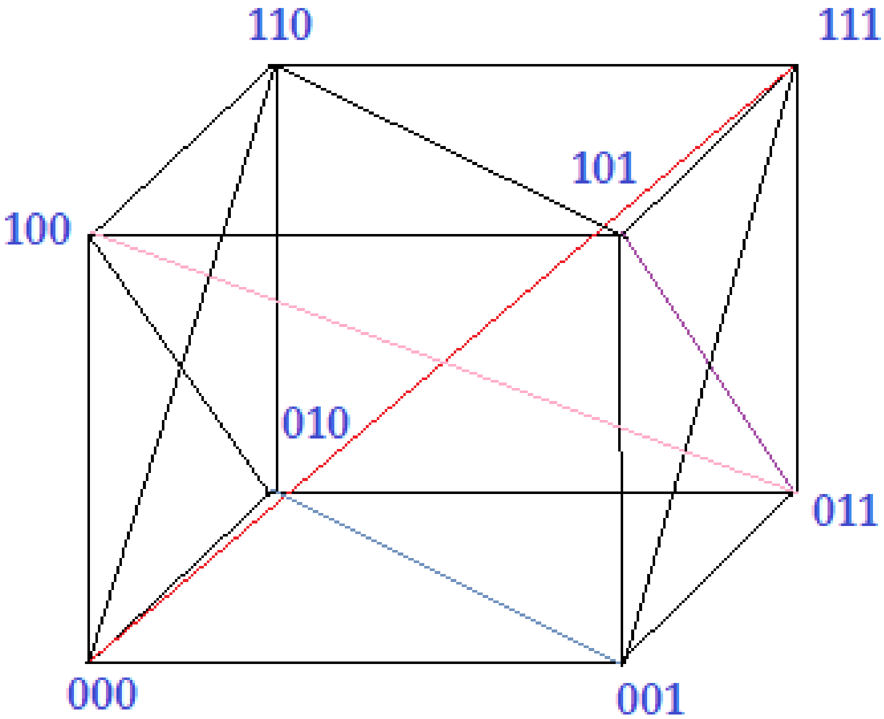
(b) There is a processor for every permutation of n distinct numbers. So, number of processors=${}^{n}P_n = n!$.

(c) We go from processor numbered $P_a = a_1 a_2 ... a_n$ to $P_b = b_1 b_2 ... b_n$. This can be done using the following algorithm:
Initialize $Path$ as P_a.
Initialize CP to P_a.
for i=n:-1:2 {
 Find b_i in CP. Say, $CP_j = b_i$.
 if $j \neq 1$, Modify CP by swapping a_1 and a_j in it.
 if $j \neq 1$,append CP in $Path$.
 Modify CP by swapping a_1 and a_i in it.
 Append CP in $Path$. }

(d) Since, we will append at most 2 processors to $Path$ at a single iteration, diameter of S_n will be $O(n)$.

4. Before proving the first statement of the problem, we first try and demonstrate how the second statement holds true. Let the h different bits between P_i and P_j be b_1, b_2,..., b_h arranged in descending order and the identical bits between P_i and P_j be B_1, B_2,..., B_{d-h}.
Path m: invert $b_m \rightarrow$ invert $b_{(m+1) \mod h} \cdots \rightarrow$ invert $b_{(m+h-1) \mod h}$
Since it takes h steps and the time when all different h bits are inverted, it is evident that there are h paths with length h and they are parallel.
For paths with length $h+2$, Path n: invert $B_n \rightarrow$ invert $b_1 \rightarrow$ invert b_2 ... $\rightarrow$ invert $b_h \rightarrow$ invert B_n.
For each such path, first inverted bit is different followed by identical steps making the $d-h$ paths parallel to each other.
Now that we have listed h parallel paths of length h and $d-h$ paths of length $h+2$, exhaustively, there can be only these d parallel paths between P_i and P_j hence proving the first statement of the problem.

5. PM2I interconnection network of 8 processors will look like:



Having way more connections, this is NOT the same as the hypercube.