CS525 - Parallel Computing
Homework 3 - jain207@purdue.edu

Q1) prod_cons_1.c

Consumption for processors:

4 - 29,383,718
8 - 32,485,111
16 - 6,272,493

Q2) prod_cons_2.c

4 - 76,146,440
8 - 104,384,192
16 - 62,040,402

Q3) prod_cons_3.c

Sample command for 8 processors:

mpicc prod_cons_3.c -o output
mpirun -n 16 -rf rankfile_8 output

where output is the executable

rankfile_4
rankfile_8
rankfile_16

4 - 3,783,491
8 - 16,714,397
16 - 16,408,479

Q4)
We will be proving this by taking the opposite case of k-to-all broadcast as the message communication will be exactly the opposite of what's required. So, the complexity will be the same for both.
This will be done on a **hypercube topology** since it is the most dense network.

In k-to-all broadcast, in the worst case, the message size doubles in each of the first log k iterations, and then remains mk in the remaining *log(p/k)* iterations.
The total communication time of the first k iterations is $t_s \log k + t_w m(k - 1)$. The total communication time of the last log(p/k) iterations is $(t_s + t_w mk) \log(p/k)$ as mk message size will be communicated in log(p/k) point to point transfers

Thus, the entire operation is performed in **[$t_s \log p + t_w m(k \log(p/k) + k - 1)$]** time.

Q5) For k-to-All scatter assuming **hypercube topology**, an all to all personalized communication takes place among k processors resulting in $[(t_s + t_w p)k-1]$ time followed by one to all personalized communication from each of these k processors to all other processors taking $t_s \log(p/k) + t_w p(p-k)$ time as each of the kth processor will be responsible for approximately p/k processors

Thus, the entire operation will be performed by combining the above 2 times resulting in
[$t_s (\log(p/k) + k - 1) + t_w p(p - 1)$] time.