CS 491/521 Parallel Programming
Homework 4: MPI
Due: 11/10, Thursday, 11:59 pm

Create an MPI based parallel implementation of the Jacobi program shown below. Evaluate performance on 4, 8, 12, and 16 processes (on 1, 2, 3, and 4 nodes, respectively).

```
for(iter=0; iter<maxiter; iter++)
 {
     maxdiff = 0;
     for(i=1; i<N+1; i++)
         for(j=1; j<N+1; j++)
         {
                 xnew[i][j] = 0.25*(xold[i-1][j]+xold[i+1][j]+xold[i][j-1]+xold[i][j+1]);
                 if ((thisdiff=fabs(xnew[i][j]-xold[i][j]))>maxdiff)
                         maxdiff = thisdiff;
         }
     if(maxdiff<epsilon)
     {
         printf("Solution converged in  %d iterations\n",iter+1);
         printf("Solution at center of grid : %f\n",xnew[(N+1)/2][(N+1)/2]);
         break;
      }
     for(i=1; i<N+1; i++)
         for(j=1; j<N+1; j++)
                 xold[i][j] = xnew[i][j];
}
```

Note that the code above is the same as the one used in the previous OpenMP assignment, i.e., you can re-use the previous template code.

Write a report about how you perform the optimization. Show and explain the results you observed. You should include the actual outputs from your runs in an Appendix. Also, for measuring the performance, you should run the code for at least 3 times and report the average performance. In addition, your report should include instructions for the TA about how to compile and execute your code on a CS departmental cluster.

Submit your report as well as your optimized code to Canvas.

For details on how to run MPI programs on the departmental cluster, please refer to TA's tutorial or contact the TA if necessary.