1. My code is implemented in this zip folder.

My code solves the poison equation of the following density and potential function:

(d0 is background potential, set to be 10.0)

I iterate the SOR method until the error reaches 1e-6.

I utilize odd-even ordering so that I can use omp parallel calculation and produce constant answer each time. (The content is within void matrix::SOR\_smoothing(const matrix &rho,double omega,int steps) in matrix.cpp)

I used different number of threads (number\_of\_threads = 1,2,3,4,5,6,7,8) and tested on different sizes of 2D array (size = 100, 200, 400). I obtained the results as followed.

I defined performance as and normalize performance (num\_threads = 1) to be 1.

From the slopes of the regression line we can see that the parallel efficiency increases with increasing array size.

One interesting thing is that all cases have best performance at num\_threads=2, instead of 4, which is the number of my CPUs. I think it may stem from some other background programs my computer is running.