CS 594 – Homework 11
due April 25th

The purpose of this homework is for you to get some hands on experience in working with sparse matrices and optimizing sparse matrix-vector product. The techniques that you would use (except if you come up with new ideas) would be from Lecture #10.

File matrix.output contains a sparse matrix in ASCII AIJ format. The first line gives 3 numbers

```
number_of_columns  number_of_rows number_of_nonzeroes
```

followed by the matrix. There are 2 parts of the assignment.

1. Write routines (and a driver to use them; in C/C++ or Fortran) that
   1. Read the matrix and store it in CRS or CCS format.
   2. Perform matrix-vector product where the matrix is in CRS/CCS format.
   3. Test that the code is correct
   4. Report Mflop/s rate using PAPI
      http://www.cs.utk.edu/~terpstra/using_papi/
      http://icl.cs.utk.edu/papi/

2. In this part you have to optimize your code. Two optimization strategies are suggested for this particular case. It is known (inside information about the matrix structure) that certain reordering may be beneficial for the performance, in particular, do the following index reordering. Old index i_{old} becomes
   $\mathbf{i_{new} = (i_{old} - 1) / 8660 + 3 * ((i_{old} - 1)\% 8660) + 1.}$
   Here '/' is integer division and '%' is the modulo operation. Note that reordering here means that if the old matrix (before reorder) had a nonzero element

i_{old}	j_{old}	$A_{2,8661}$		i_{new}	j_{new}	$A_{4,2}$
2	**8661**	**4.5**	the new matrix will have	**4**	**2**	**4.5**

   1. Report the Mflop/s mat-vec rate with the reordered matrix. Is it faster than the mat-vec product with the original matrix and why?
      Hint: plot the nonzero structures of the original and reordered matrices and compare.
   2. Save the reordered matrix in BCRS format with blocks of size 3x3. Report again the Mflop/s rate and compare with the other 2 cases.

   Bonus points: can you think of other ways to optimize the performance?