5 (10 points) Write an input generator which creates completely random preference lists, so that each M has a random permutation of the W's for preference, and vice-versa. The purpose of this problem is to explore how "good" the algorithm is with respect to M and W. (There is an interesting meta-point relating to algorithm fairness that can be made with this problem.)

We define "goodness" of a match as the position in the preference list. We will number positions from one (not zero as is standard for array indexing.) Note that lower numbers are good. To be precise, suppose m is matched with w. The $mRank$ of m (written $mRank(m)$) is the position of w in m's preference list, and the $wRank$ of w is the position of m in w's preference list. We define the $MRank$ of a matching to be the sum of all of the $mRank(m)$ and the $WRank$ of w to be the sum of all of the $wRank(w)$. If there are n M's (and n W's), we define the $MGoodness$ to be $MRank/n$ and the $WGoodness$ to be $WRank/n$.

As the size of the problem increases - how does the goodness change for M and W? Submit a short write up about how the goodness varies with the input size based on your experiments. Is the result better for the M's or W's? You will probably need to run your algorithm on inputs with n at least 1,000 to get interesting results.

After running the test for different input size N, I observed the following:

- As the size of the problem increase, the goodness increases for both M, and W.
- The M_Rank increases in logarithm with respects to input size.
- The M always have a better goodness (Smaller M_Rank) than W.

Tests Data:

M Rank	W Rank	N
4.9205	25.2448	125
6.0083	41.9761	250
6.6596	76.6302	500
7.4456	136.6873	1000
8.0895	253.5886	2000
8.9198	456.4240	4000
9.7359	837.9548	8000

Methodology:

We run the G-S algorithm on randomly generated permutation matrix. We repeat it 50 times for each of the problem size and then take the average for the M_Rank and W_Rank.

Observations:

After putting then into MATLAB, I am also able to confirm that the $\log N \propto Y$. Another observation made was that, the product of M_Rank and W_Rank is approximately the problem size but slightly higher.