I randomized every single choice and ran it multiple times. I used this strategy and as an "optimiziation" because
I found out that it doesn't take a long time to run through 1~600 files. Thus, it would be more optimal in terms of
time spent on coding to make it randomized choice and run it for long time rather take long time to write a complicated
but semi-optimal code and hoping that it would work. After running it multiple times, I then checked output files for all answers that produced only one relay team. Output that produces one relay team MUST be an optimial answer for that problem. Thus, I removed those input files from running 1~600 problems, therefore, less files to run per loop. Meanwhile, other group members used Tarjan algorithm which find strongly connected components to generate output file. We then merged those two output files based on higher scores generated. Randomizer algorithm favors small input files and Tarjan favors big input files.