1. Suppose the time that Java takes to sort a 1,000,000 length array is approximately $J \sim \mathcal{N}(\mu = 46, \sigma^2 = 6^2)$ milliseconds (ms), since it uses the (randomized) QuickSort Algorithm.
   a. Python initially implements a (deterministic) MergeSort Algorithm, and it always finishes in $P = 49$ ms. What is the probability that Java sorts a single 1,000,000 length array faster than Python does? Show your work and give your answer rounded to 4 decimal places.
   b. Python attempts to implement QuickSort as well, but did it less efficiently. Its runtime is approximately $P \sim \mathcal{N}(\mu = 55, \sigma^2 = 8^2)$. What is the probability that Java sorts a single 1,000,000 length array faster than Python does? Show your work and give your answer rounded to 4 decimal places.
   c. The remaining parts are left for you ☺.

Solution: Watch lecture ☺