Size,Base Combinations v. Time

- max_time = 120

- As can be seen in the graph, regardless of table size, a base of 1 is a terrible choice and drastically degrades performance. This is due to the fact that 1 is a factor of every integer value. Maximizing collision probability.

- As observed in 250727, 250726, when the chosen base approaches the table size, performance is also degraded. This is due to the uniform hash function. A base which is close to the table size will result in little variance of the modulus function, increasing collisions in the hash table.

- The remaining results show that with a sufficiently large base – which is reasonably different from the table size – the main factor improving performance is the table size. Increasing table size shows small performance increases. This is because the probability of collisions is greatly reduced as the number of available 'slots' increases.

Overall, the optimal hash table utilizes a large base which varies from the table size by a large factor. It'll also implement a table size far in excess of the number of pairs required as this also aids in the decrease of collisions and clustering.