Question 2 Report

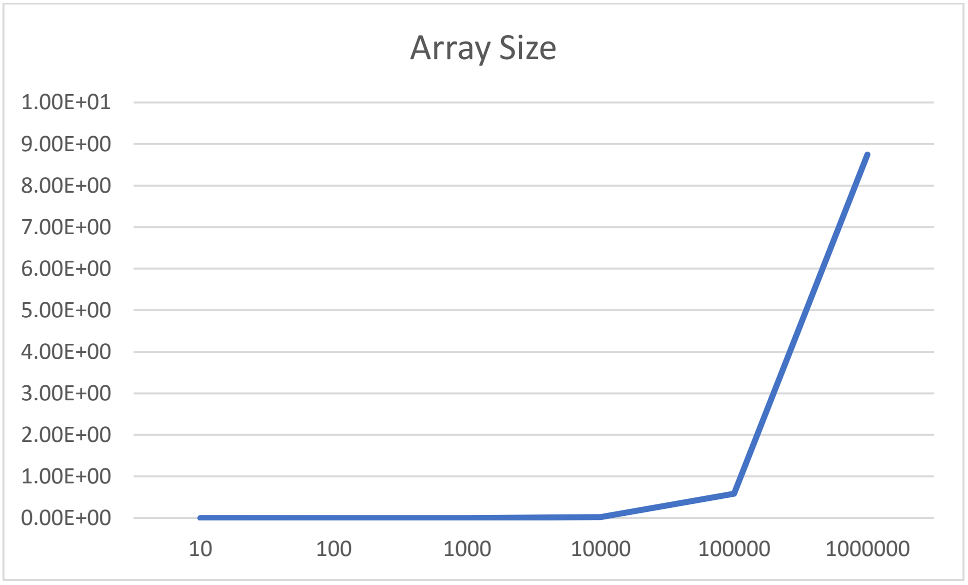
b) The merge sort time complexity is nlog(n) while the binary search is log(n). The algorithm I created uses both the merge sort and then the binary search is used in a loop. Therefore, the time complexity of the algorithm is also nlog(n).

T(n) = complexity of merge-sort + complexity of the for loop with binary search inside

T(n) = O(nlog(n)) + O(nlog(n)) = 2 * O(nlog(n))

T(n) = O(nlog(n))

c)



I experimented with values ranging from 1 to 1000000. That resulted to this graph which is approximately the same as nlog(n) where larger values of n result in a much longer time to compute the getPairs function.