Mubarra Khan
Z5161249
Q1 solution

a) First we will go through all of $\binom{n}{2} = \frac{n(n-1)}{2}$ pairs $(A[k], A[m]), k < m,$ of distinct integers in A; and compute the sums $A[k]^2 + A[m]^2 \quad forall \quad 1 \leq k < m \leq n.$

This is achieved using two for loops where the first for loop will be $forall \;\; 0 \leq i < len(A)$ and the second loop will be between $forall \;\; i + 1 \leq j < len(A)$ so the same pair is not counted twice. Since the numbers in A are distinct we can be sure there will be no distinct pairs. This will run in $O(n^2)$ time.
To store the sums computed , a binary tree will be used which has a run time of $log(n)$. At each iteration the sum will be stored in the tree and if the sum already exists in the tree it will mean we have found two different pairs that have the same sum. Overall this algorithm will have $O(n^2 log(n)$) run time.

b) For this part we will take the same approach as the previous part for computing the sum of distinct pairs having a run time of $O(n^2)$. To store this however we will use a hashMap which has a run time of $O(1)$. At each iteration we perform to compute the sum, we will store it in a hash map and if a duplicate key exists it will mean we have found duplicate pairs with the same sum.