Q1.

(a)

First go through all of $\frac{n*(n-1)}{2}$pairs of tuples of (A[k] , A[m]) where k < m of distinct integers in array A. Then allocate an integer array of size $\frac{n*(n-1)}{2}$. After that, compute the sum of each tuple produced(A[k]^2 + A[m]) in the previous phase and store sums into the allocated array. Then store the combination of(A[k], A[m]) using a hash table, the hash key of the tuple will be A[k]^2 + A[m](takes $\frac{n*(n-1)}{2}$). Using advanced sorting algorithms which usually take O(n*logn)(e.g Quick sort,merge sort), so the array can be sorted in O(n^2 * logn^2) = O(2*n^2 * logn) = O(n^2 * logn). Go though the sorted array and try to determine any duplicates in the array. If there exists duplicates of the sums, then use this value to find the tuples in the hash table, if there exists two tuples(a,b),(c,d) corresponding to the sum, and all a b c d are distinct,then there exists four distinct

integers that satisfy the equation m^2 + s = k^2 + p.

Time complexity:
$\frac{n*(n-1)}{2}$(n^2, find all the combinations) + 2* $\frac{n*(n-1)}{2}$(n^2,store the sums into the array, and create hash table) + 2*n^2 * logn(n^2 * logn, sort the array of sums) + (n+k) (determine any duplicates), therefore the overall time complexity should be O(n^2 * logn)

(b)
Use a hash table of size $\frac{n*(n-1)}{2}$(to avoid the case (a,b),(b,a))for all 1 ≤ i < j ≤ n hashing all the m^2 + s. And hash all $\frac{n*(n-1)}{2}$tuple using the method introduced in the previous part.Then go through all of the n 2 slots and check by brute force if the same number appears twice. If there exists duplicates of the sums, then use this value to find the tuples in the hash table, if there exists two tuples(a,b),(c,d) corresponding to the sum,

and all a b c d are distinct,then there exists four distinct integers that satisfy the equation m^2 + s = k^2 + p.

Time complexity:

$\frac{n*(n-1)}{2}$(n^2, find all the combinations) + 2* $\frac{n*(n-1)}{2}$(n^2,store the sums into the array, and create hash table) + n^2(check by brute force) + k(determine duplications in the tuples)