JIE MEI z5173405

Question 1

Firstly, I need to calculate all possible sums of squares in array A. The algorithm I designed is implemented using two loops. The first layer loop sets a variable i from 0 to n, and the second layer loops sets a variable j from i+1 to n. The time complexity of this algorithm is O(n^2).

Part A:

I will use an array B to store all possible sums of squares. Sort B using merge sort. Then we traverse the entire array to find the elements of the same value in the array. The time complexity should be O(n^2).

Sample code:

```
int A[n];
int B[n(n-1)/2];
int z = 0;
for(int i = 0; i < n; i++) {
    for(int j = i+1; j < n; j++) {
        B[z] = (A[i])^2 + (A[j])^2;
        z++;
```

```
        }
    }
```

Part B:

We will use hash table to store all the sum of squares. Like storage with arrays, the hash table needs to be inserted in each iteration and then check whether the data is successfully inserted. The time complexity of this algorithm is O(n^2).