Online Assignment 1

Duration: 1h

Notes: If you have to use basic sort algorithms for coding: Insertion sort, Merge sort, Quick sort, Bin sort, you don't need to write it in detail, just call it. We assume that these basic algorithms already implemented.

Question 1(3 points): Find the time complexity of the following function (Big-Oh). You only need to write down answers, don't need to write explanation.

a.

```
function( int n ) {
        if(n == 1) return;
        for(int i = 1 ; i <= n ; i + + ) {
                for(int j= 1 ; j <= n ; j + + ) {
                        printf("*" );
                        break;
                }
        }
}
```

b.

```
void function(int n) {
        int i, j, k , count =0;
        for(i=n/2; i<=n; i++)
                for(j=1; j<=n; j= 2 * j)
                        for(k=1; k<=n; k= k * 2)
                                count++;
}
```

c.

```
function( int n ) {
        if( n == 1 ) return;
        for(int i = 1 ; i <= n ; i + + )
                for(int j = 1 ; j <= n ; j + + )
                        printf("*");
        function( n-3 );
}
```

Question 2 (3 points) Find the time complexity of T(n). You only need to write down answers, don't need to write explanation.

a. $T(n) = 3T(n/2) + n$
b. $T(n) = 3T(n/3) + n/2$
c. $T(n) = 2T(\sqrt{n}) + \log n$

Question 3 (2 points): Given the following sorting algorithm

```
void new_Sort(int array[], int n) {

  for (int step = 0; step < n - 1; ++step) {

    for (int i = 0; i < n - step - 1; ++i) {

      if (array[i] > array[i + 1]) {
        int temp = array[i];
        array[i] = array[i + 1];
        array[i + 1] = temp;
      }
    }
  }
}
```

Between the above algorithm and Insertion Sort, which one is better for sorting? Explain your answer.

Question 4 (2 points): Write a fastest program which finds the number that appeared the maximum number of times in an array. You aren't allowed to use a temporary array or extra spaces but you can use recursive function or few temporary variables. After writing, please also show the time complexity of your program

For example: array [] = 1,2,4,18,2,10,3,18,2 -> output is 2