Once you have completed your solution for each problem, perform a run time analysis (Worst Case Big-O Notation) of your solution. Document this for each problem and include this in your submission.

TASK0

1. Csv.reader reads line by line in the file, hence the complexity is O(n) – for one file -> n is the number of lines in the file
2. O(1) for print the topmost line

**Since there is one “calls” file and one “texts” file- the complexity is O(2)+2**

TASK1 – Implementation 2nd way

1. The reading of files has complexity of O(2n)
2. For loop is executed depending on the rows in the file: O(n) -> n is the number of rows in the file. There are two different loops : hence complexity is O(2n)
3. Let’s take the worst case and say the if condition is executed every time- which means all the numbers are distinct , then in this case complexity is O(n)
4. For calls , two if conditions have two if conditions – hence O(2n)
5. Appending is O(1)- for worst case- will be appending n times , O(n) -> O(2n)
6. Printing the output is O(1)

**Hence for call records: total complexity -O(4n)**

**Hence for text records: total complexity- O(4n)**

**Total complexity: O(8n+1)**

TASK2

1. Reading of the file is dependent on the size of the file, reads every line by line: O(n)
2. For loop to iterate through calls list: O(n)
3. Either if or else condition will be executed: then this is executed once in one iteration, then O(n)
4. Addition of the number inside the if-else condition is also once per iteration: so O(1) for one addition; for n iterations : O(n)
5. Calculating maximum takes : O(1)
6. Printing the output takes: O(1)

Except step 1,5,6 , rest are same for receiving numbers as well.

Hence for reading the file , calling numbers and receiving numbers together : O(n) + O(3n)+ O(3n )

**So total: O(7n+2)**

TASK3

1. Reading of the file is dependent on the size of the file, reads every line by line: O(n)
2. For loop is executed n times : O(n)
3. O(2) for search of the pattern in every iteration of loop: (2n)
4. If condition in worst case will be executed once in every iteration, 4 if conditions: O(3n)
5. Summing up in order to get the Bangalore calls that are received: O(n)
6. Dictionary is used to maintain the key value pairs and maintain the uniqueness, assigning the dictionary with key and value is executed once for if-else condition
7. Printing the output is O(1)

**Hence: O(9n+1)**

TASK4

1. Reading of the file is dependent on the size of the file, reads every line by line: O(n)
2. For loop is executed n times : O(n)
3. If condition in worst case will be executed once in every iteration and so will the appending of the number to the telemarketer list
4. Hence O(n) and O(n) will be complexity for two if conditions
5. And O(n) and O(n) for appending or removing the elements from the telemarketers in the list
6. O(1) for printing the output
7. This steps 1 to 4 also holds good for the texts csv file.

**Hence , total complexity is O(11n+1)**