Benjamin Parrish, cbe982
CS 2123-003
9/17/2019

Problem 1

```
1   sum = 0;        //1
2   i= 4;           //1
3   while ( i> 3 && i< 10) {    //will run 7 times checking i =[4, 10
4         sum += A[ i];         //4 and 5 run 6 times each because the wile loop cycles 6 times
5         i++;
6   }
```

1+1+7+6+6 = 21

There are 21 instructions in this program.

Problem 2

```
1   sum = 0;        //1
2   for ( i=0; i< numElements;i++ ) { //will execute numElements+1 times
3     sum += A[ i];                   //will execute numElements times
4   }
```

The number of instructions, in terms of nunElements is:
1+numElements+1+numElements = 2(numElements +1)
It can be reduced to n:

Problem 3

```
1   sum = 0;              // 1
2   for ( i=0; i< n; i++ ) {   //will run n+1 times
3     for (j=0; j < n; j++) {  //will run n*(n+1) times
4        sum += i* j;          //will run n^2 times
5     }
6   }
```

The number of of instructions, in terms for in is:
$n^2 + n^2 + n + n + 1 + 1 = 2(n^2 + n +1)$
It can be reduced to n^2