Marinel Tinnirello
CMPT 435 - Assignment #2

1)
```
for (int i = n; i >= n; i = i/2) {          // n / 2 (execute (n / 2) times)
  // some O(1) time statements;             // 1
}                                           // 1 * (n / 2)
```
O(n) = n / 2

2)
```
for (int i = n; i > 0; i /= 2) {            // n / 2 (execute (n / 2) times)
  for (int j = 1; j < n; j *= 2) {          // 2^n (execute (log n) times)
    for (int k = 0; k < n; k += 2) {        // n / 2 (execute (n / 2) times)
      // constant # of operations           // 1
    }                                       // (log n)^2 * (n / 2)
  }
}
```
O(n) = (log n)^2 * n

3)
```
for (int i = n; i > 0; i--) {               // n (execute n times)
  for (int j = 1; j < n; j *= 2) {          // 2^n (execute (log n) times)
    for (int k = 0; k < j; k++) {           // n (execute n times)
      // constant # of operations           // 1
    }                                       // 2^log n
  }
}
```
O(n) = n^2

4)
```
int j = 1, i = 0;                           // 2
while (i < n) {
  i = i + j;                                // 2 * n (execute sqrt(n) times)
  j++;                                      // 2 * n (execute sqrt(n) times)
}                                           // 2 * sqrt(n)
```
O(n) = sqrt(2n)

5)

a) Let some variable “count” represent the count for the majority element. Let some variable “element” represent the majority element.

1) check if the majority element exist
2) if the count equals 0, set the element to the index of the array
3) if the index of the array equals the element, add to the count, else decrement the count
4) reset the count to 0
5) find the majority element
6) **Output:** the majority element/lack thereof

b) **Input:** array

```
for ( int i = 0; i < n; i++ ) {
```

```
        if  ( count == 0 ) { Output: element = A[i] }
        if ( A[i] == element ) { Output: count++ }
        else { Output: count-- }
}
count = 0
for ( int i = 0; i < n; i++ ) {
        if ( A[i] == element ) { Output: count++ }
}
if ( count > (n / 2) ) { Output: element }
```