hitbullseye

Time Complexity - 2 - Functions Related
(Finding time complexity for Recursive Programs)

1. What is the time, and space complexity of the following code:

```
int a = 0, b = 0;
for (i = 0; i < N; i++) {
        a = a + rand();
}
for (j = 0; j < M; j++) {
        b = b + rand();
}
```

2. What is the time, and space complexity of the following code:

```
int a = 0;
for (i = 0; i < N; i++) {
        for (j = N; j > i; j--) {
                a = a + i + j;
        }
}
```

3. What is the time, and space complexity of the following code:

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

4. What is the time complexity of the following code:

```
int a = 0, i = N;
while (i > 0) {
        a += i;
        i /= 2;
}
```

5. What is the time complexity of the following code:

```
for(var i=0;i<n;i++)
        i*=k
```

6. What is the time complexity of the following code:

```
int value = 0;
for(int i=0;i<n;i++)
        for(int j=0;j<i;j++)
        value += 1;
```

7. Find the complexity of the below recurrence:

$$T(n) = \begin{cases} 3T(n-1), & \text{if } n>0, \\ 1, & \text{otherwise} \end{cases}$$

8. Find the complexity of the recurrence:

$$T(n) = \begin{cases} 2T(n-1) - 1, & \text{if } n>0, \\ 1, & \text{otherwise} \end{cases}$$

hitbullseye

9. Find the complexity of the below program:

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

10. Find the complexity of the below program:

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