Aufgabe 1:

1.

c(n) = n-1 + c(n/2) + c((n+1)/2) for n > 1, c(0) = c(1) = 0

|  |  |
| --- | --- |
| n | c(n) |
| 0 | 0 |
| 1 | 0 |
| 2 | 1 |
| 3 | 3 |
| 4 | 5 |
| 5 | 8 |
| 6 | 11 |
| 7 | 14 |
| 8 | 17 |
| 9 | 21 |
| 10 | 25 |

2.

m(n) = m(n-1) \* n + (n-1)! \* (n-1) \* (n+4)/2 for n > 0, m(0) = 0

|  |  |
| --- | --- |
| n | m(n) |
| 0 | 0 |
| 1 | 0 |
| 2 | 3 |
| 3 | 23 |
| 4 | 164 |
| 5 | 1252 |
| 6 | 10512 |
| 7 | 97344 |
| 8 | 990432 |
| 9 | 11010528 |
| 10 | 132966720 |

Aufgabe 2:

1.

Es wird so oft aufgerufen wie n > h[index] ist:

Erläuterung:

Zähler := 0

Index := 0

n := 20 (Distanzen: 1, 3, 4, 13, 40, …)

n > 1 (Zähler und Index um 1 erhöhen)

n > 4 (Zähler und Index um 1 erhöhen)

n > 13 (Zähler und Index um 1 erhöhen)

n < 40 -> fertig

insertionSorth wird insgesamt 3x aufgerufen.

2.

Best case O(n – h) Worst case O(n2 – h)