Aufgabe 1:

1.

c(n) = if(n < 2, 0, n-1 + c(n\2) + c((n+1)\2))

|  |  |
| --- | --- |
| 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, a(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.

Aufrufe von insertionSort = a, Anzahl an Elemente im Array = n

Distanzen sind h = 3 \* hmax + 1

a(n) = n/3

2.

Best case O(f1,5) Worst case O(f2)

f(n, h) = 3 \* h + n for n > 3