**离散卷积定义：**

**把已知函数x换成未知函数y**：

现在只有h是已知函数，初始y(0)=1已知，如何求未知函数y的所有值? 除了暴力求解O(n\*n)复杂度之外，有两种小于O(n\*n)复杂度的方法：

## 分治FFT法

对于每一个y(x)，不能重新的去算一次FFT，因为那样的话复杂度会退化成O(n^2logn),还不如暴力.

考虑找贡献，设mid=,假设我们已经知道了t∈[0,mid]的y(t),就可以计算它们对i∈[mid+1,n],y(i)的贡献,对于一个已知的y(x),对后面每个y(i)的贡献为：y(x)h(i-x-1)

如果只考虑t∈[mid+1,n]对y(t)的贡献：

而原式子是：

对于i∈[mid+1,t-1], 这部分y(i)未知, 如果我们看做0，把这个数组求卷积，得到的结果是前半部分对y(t)的贡献。只要再求i∈[mid+1,t-1]时对y(t)的贡献即可。

* 这里注意，假如已知t∈[0,mid]的y(t)，可以直接用fft算出y(mid+1)的值

**用分治思想：**先求前一半，再求它对后一半的贡献。由于先算左再算右，一定最先递归到y(0)，而y(0)已知

1. y(0)已知，当递归到y(0)时，可以直接求单个数的卷积得到y(1)；
2. 现在已知y(0,1)，用y(0)和y(1)算出它们对y(2)和y(3)的贡献，其中y(2)可以直接得到，再算y(2)对y(3)的贡献，补全y(3)；
3. 现在已知y(0…3)，可以求出它们对t∈[4,7]的y(t)的贡献，其中y(4)可以直接得到,然后递归t∈[4,7]，用类似的方法，即：用y(4)补全y(5),用y(4)和y(5)求出它们对y(6)和y(7)的y(t)的贡献,此时y(6)可以直接得到，最后用y(6)补全y(7)
4. 现在已知y(0…7)，可以求出它们对t∈[8,15]的y(t)的贡献…

以上就是分治fft中分治的思想，每次算贡献都是用fft求卷积，时间复杂度为O(n\*)

## 生成函数法

\