Definition

A recurrence relation is an equation that recursively defines a sequence where the next term is a function of the previous terms (Expressing Fn as some combination of Fi with $i<n$)

Example − Fibonacci series − $Fn=Fn-1+Fn-2$,

Tower of Hanoi − $Fn=2Fn-1+1$

Problem no:1

The Tower of Hanoi A popular puzzle of the late nineteenth century invented by the French mathematician Édouard Lucas, called the Tower of Hanoi, consists of three pegs mounted on a board together with disks of different sizes. Initially these disks are placed on the first peg in order of size, with the largest on the bottom (as shown in Figure 2). The rules of the puzzle allow disks to be moved one at a time from one peg to another as long as a disk is never placed on top of a smaller disk. The goal of the puzzle is to have all the disks on the second peg in order of size, with the largest on the bottom. Let Hn denote the number of moves needed to solve the Tower of Hanoi problem with n disks. Set up a recurrence relation for the sequence $\{Hn\}$.

Solution: Begin with n disks on peg 1.We can transfer the top $n - 1$ disks, following the rules of the puzzle, to peg 3 using $Hn-1$ moves (see Figure 3 for an illustration of the pegs and disks at this point). We keep the largest disk fixed during these moves. Then, we use one move to transfer the largest disk to the second peg. We can transfer the $n - 1$ disks on peg 3 to peg 2 using $Hn-1$ additional moves, placing them on top of the largest disk, which always stays fixed on the bottom of peg 2. Moreover, it is easy to see that the puzzle cannot be solved using fewer steps. This shows that.

$H_n = 2H_{n-1} + 1$.

The initial condition is $H_1 = 1$, because one disk can be transferred from peg 1 to peg 2, according to the rules of the puzzle, in one move.

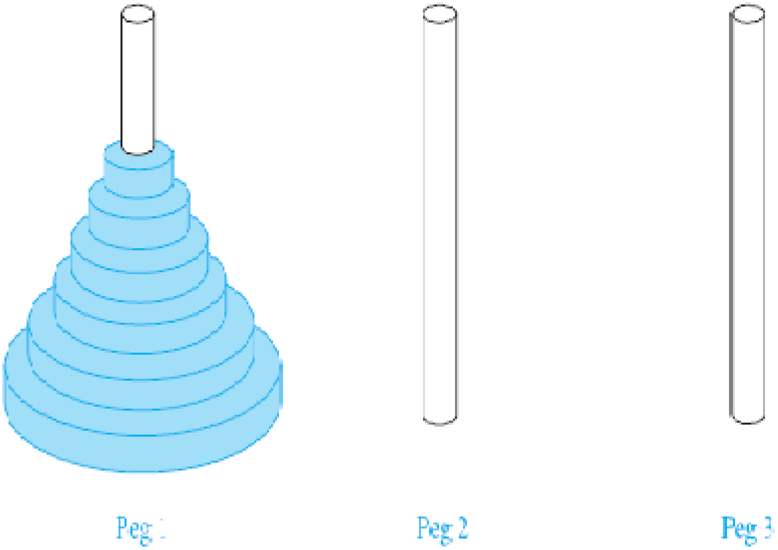


FIGURE 2 The Initial Position in the Tower of Hanoi.

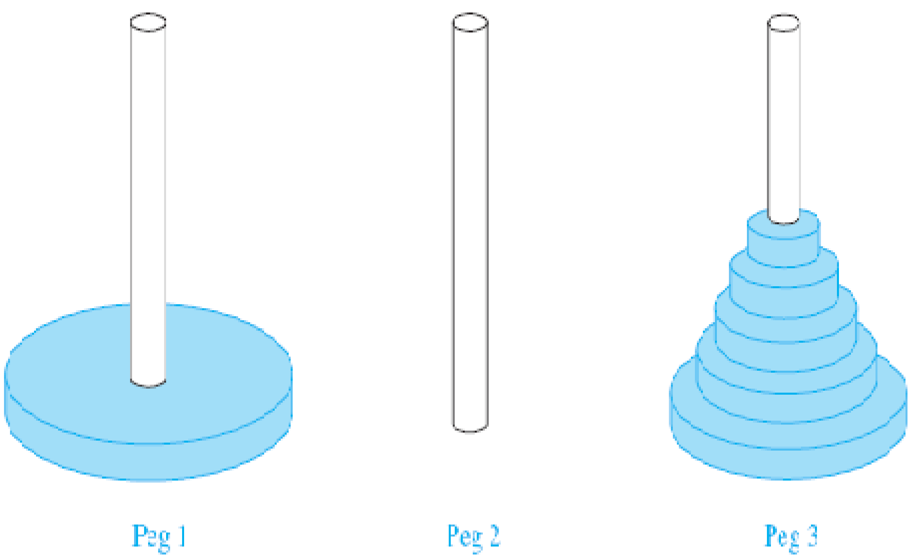


FIGURE 3 An Intermediate Position in the Tower of Hanoi.

We can use an iterative approach to solve this recurrence relation. Note that

$H_n = 2H_{n-1} + 1$

$= 2(2H_{n-2} + 1) + 1 = 2^2H_{n-2} + 2 + 1$

$= 2^2(2H_{n-3} + 1) + 2 + 1 = 2^3H_{n-3}$

$+ 2^2 + 2 + 1$

$\ldots$

$= 2^{n-1}H_1 + 2^{n-2} + 2^{n-3} + \cdots + 2 + 1$

$= 2^{n-1} + 2^{n-2} + \cdots + 2 + 1$

$= 2^n - 1.$

Problem no:2

Find a recurrence relation and give initial

conditions for the number of bit strings of length n

that do not have two consecutive 0s. How many

such bit strings are there of length five?

Solution: Let a_n denote the number of bit strings of length n that do not have two consecutive 0s.
To obtain a recurrence relation for $\{a_n\}$,
note that by the sum rule, the number of bit strings of length n that do not have two consecutive 0s equals the number of such bit strings ending with a 0 plus the number of such bit strings ending with a 1.
We will assume that $n \geq 3$, so that the bit string has at least three bits.
The bit strings of length n ending with 1 that do not have two consecutive 0s are precisely the bit strings of length $n - 1$ with no two consecutive 0s with a 1 added at the end. Consequently, there are a_{n-1} such bit strings.
Bit strings of length n ending with a 0 that do not have two consecutive 0s must have 1 as their $(n - 1)$st bit; otherwise they would end with a pair of 0s. It follows that the bit strings of length n ending with a 0 that have no two consecutive 0s are precisely the bit strings of

length $n - 2$ with no two consecutive 0s with 10 added at the end. Consequently, there are a_{n-2} such bit strings.

We conclude, as illustrated in Figure 4, that

$a_n = a_{n-1} + a_{n-2}$

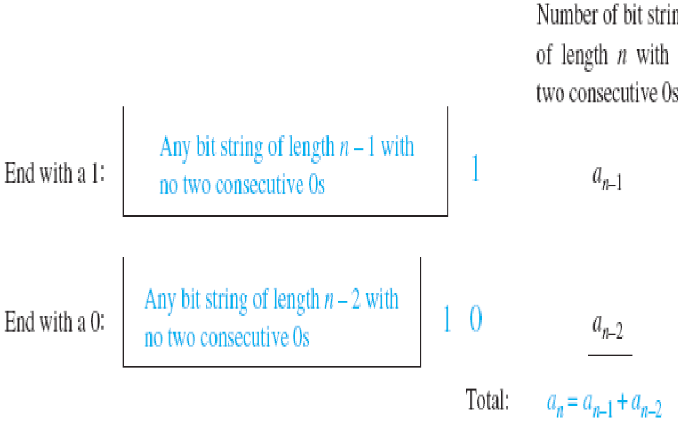
for $n \geq 3$.

The initial conditions are $a_1 = 2$, because both bit strings of length one, 0 and 1 do not have consecutive 0s, and $a_2 = 3$, because the valid bit strings of length two are 01, 10, and 11. To obtain a_5, we use the recurrence relation three times to find that

$a_3 = a_2 + a_1 = 3 + 2 = 5,$

$a_4 = a_3 + a_2 = 5 + 3 = 8,$

$a_5 = a_4 + a_3 = 8 + 5 = 13.$



Problem no:3

Codeword Enumeration A computer system considers

a string of decimal digits a valid

codeword if it contains an even number of 0 digits. For instance, 1230407869 is valid, whereas 120987045608 is not valid.

Let a_n be the number of valid n-digit codewords. Find a recurrence relation for a_n.

Solution: Note that $a_1 = 9$ because there are 10 one-digit strings, and only one, namely, the string 0, is not valid.A recurrence relation can be derived for this sequence by considering how a valid n-digit string can be obtained from strings of $n - 1$ digits. There are two ways to form a valid string with n digits from a string with one fewer digit. First, a valid string of n digits can be obtained by appending a valid string of $n - 1$ digits with a digit other than 0. This appending can be done in nine ways. Hence, a valid string with n digits can be formed in this manner in $9an-1$ ways. Second, a valid string of n digits can be obtained by appending a 0 to a string of length $n - 1$ that is not valid. (This produces a string with an even number of 0 digits because the invalid string of length $n - 1$ has an odd number of 0 digits.) The number of ways that this can be done equals the number of invalid $(n - 1)$-digit strings. Because there are $10n-1$ strings of length $n - 1$, and $an-1$ are valid, there are $10n-1 - an-1$ valid n-digit strings obtained by appending an invalid string of length $n - 1$ with a 0. Because all valid strings of length n are produced in one of these two ways, it follows that there are

$$an = 9an-1 + (10n-1 - an-1)$$
$$= 8an-1 + 10n-1$$

valid strings of length n.