

Handshake Problem – Time Complexity

Using substitution method

As we see the program , the return value is: return handshake($n - 1$) + ($n - 1$) , there is two approaches:

First Approach:

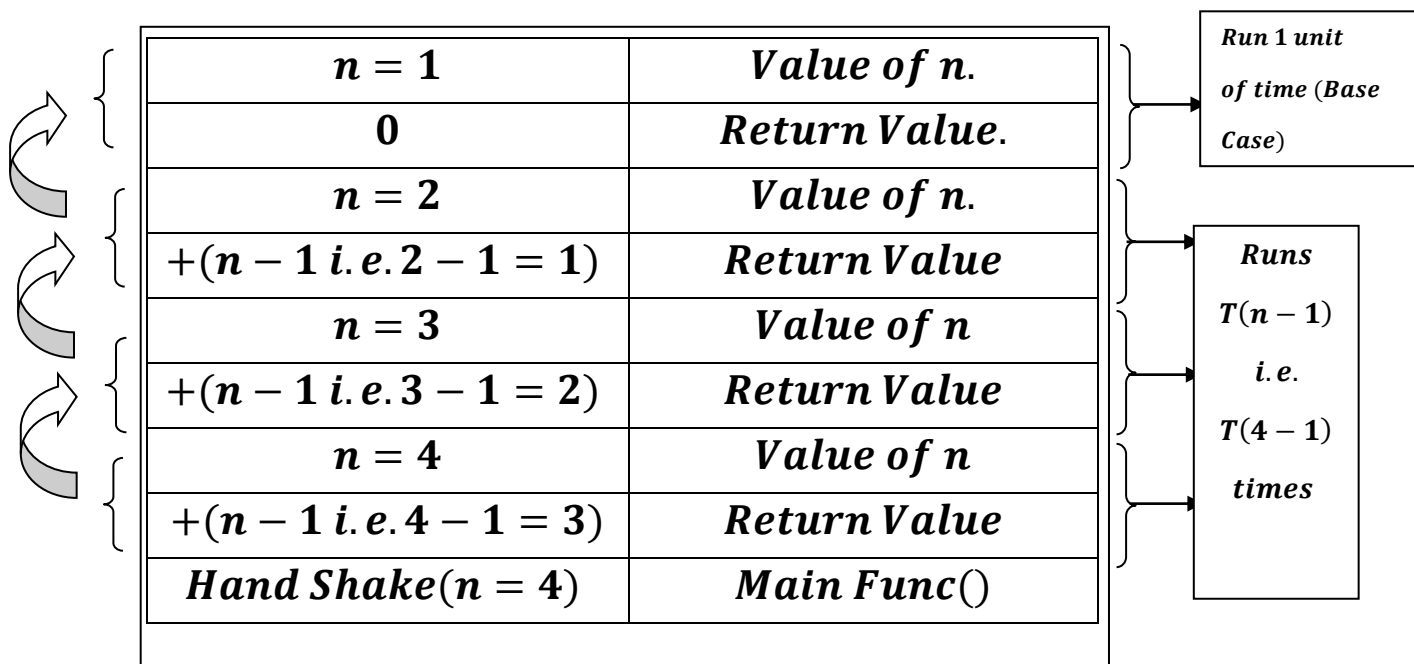
Take $n = 4$

Hence Base Cases:

$n = 0$, has 0 handshake and it returns 0 takes 1 unit of time.

$n = 1$, has 0 handshake and it returns 0 takes 1 unit of time.

When $n > 1$, here 4, therefore total push inside stack:

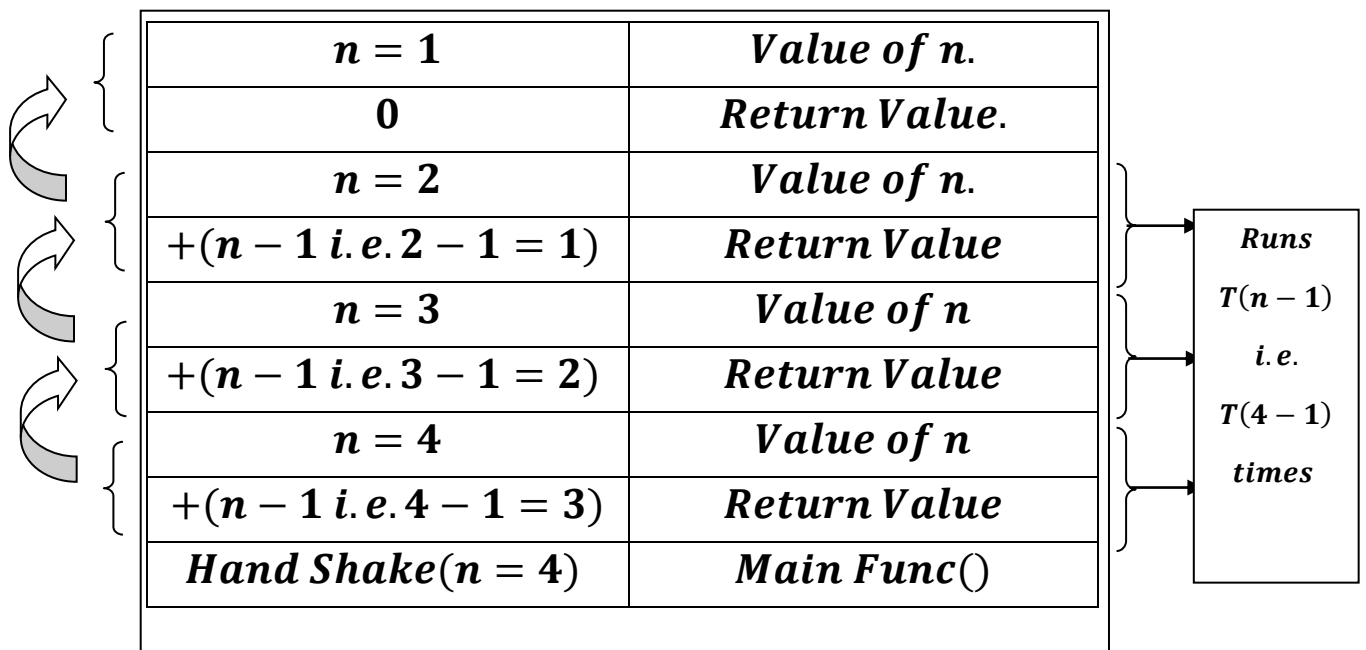


Hence we get our relation:

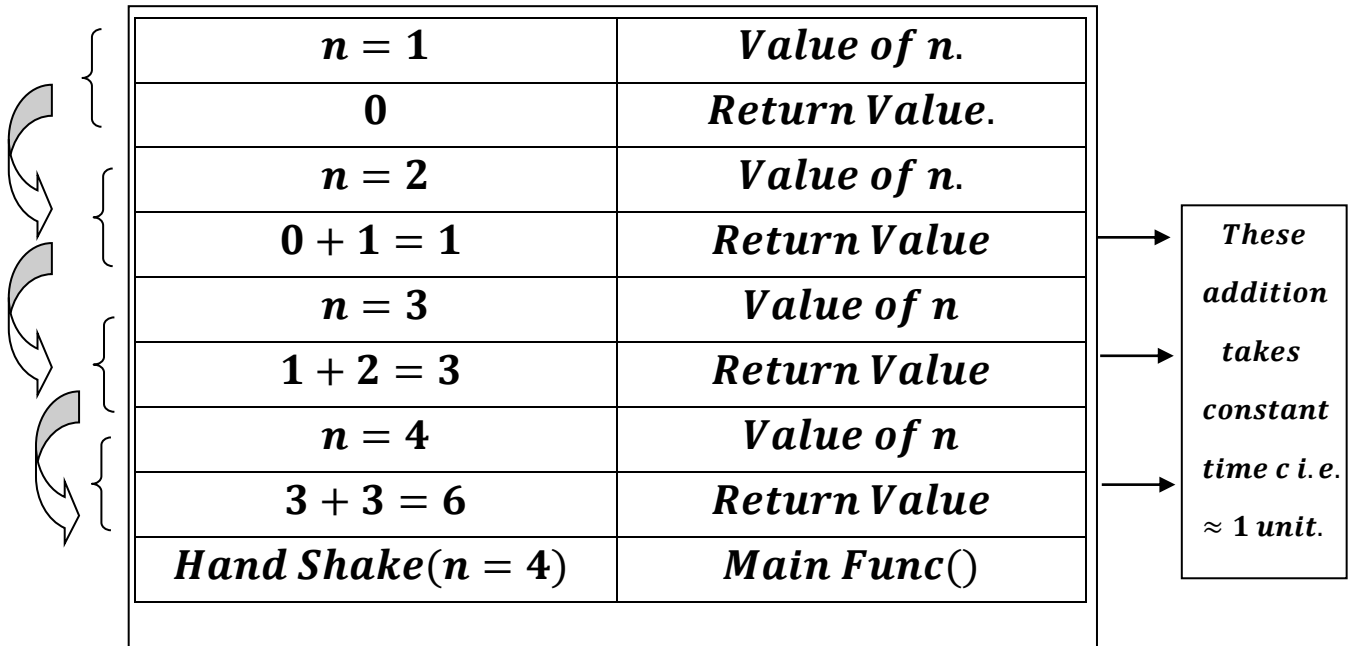
$$T(n) = \begin{cases} 1 & , \text{for } (n = 0) \\ 1 & , \text{for } (n = 1) \\ T(n - 1) + T(1) & , \text{for } (n > 1) \end{cases}$$

2nd Approach:

For base cases remain same , Lets take the stack representation of push:



Lets take the addition operation in pop:



$$T(n) = T(n - 1) + 1, \text{ for } n > 1$$

To Compute
 $H(n - 1)$

To add
 $H(n - 1)$ by
 $n - 1$

Which gives:

$$T(n) = \begin{cases} 1 & , for (n = 0) \\ 1 & , for (n = 1) \\ T(n - 1) + 1 & , for (n > 1) \end{cases}$$

Moving with the 1st way:

$$T(n) = \begin{cases} 1 & , for (n = 0) \\ 1 & , for (n = 1) \\ T(n - 1) + T(1) & , for (n > 1) \end{cases}$$

And we know $T(1) = 1$, hence rewriting the linear recurrence equation as :

$$T(n) = T(n - 1) + 1$$

Or 2nd way:

$$T(n) = \begin{cases} 1 & , \text{for } (n = 0) \\ 1 & , \text{for } (n = 1) \\ T(n - 1) + 1 & , \text{for } (n > 1) \end{cases}$$

i. e. same $T(n) = T(n - 1) + 1$

Therefore , continuing with the equation: $T(n - 1) + 1$,

$$T(n - 1) = T(n - 1 - 1) + 1 = T(n - 2) + 1$$

Substituting this in $T(n)$ we get:

$$T(n) = (T(n - 2) + 1) + 1$$

$$\text{Now , } T(n - 2) = T(n - 2 - 1) + 1 = T(n - 3) + 1$$

Substituting this in $T(n)$ we get:

$$T(n) = ((T(n - 3) + 1) + 1) + 1$$

Now if it runs upto `i` times we get :

$$\text{Now , } T(n - i) = T(n - i - 1) + 1$$

Substituting this in $T(n)$ we get:

$$T(n) = T(n - i - 1) + 1 + 1 + 1 + \dots i \text{ times}$$

When $i = n - 1$ we get:

$$T(n) = T(n - (n - 1) - 1) + 1 + 1 + 1 + \dots (n - 1) \text{ times}$$

$$= T(n - n + 1 - 1) + 1 + 1 + 1 \dots (n - 1) \text{ times}$$

$$= T(0) + 1 + 1 + 1 \dots (n - 1) \text{ times}$$

And we know $T(0) = 1$ along with $T(1)$ is 1 and as 1 is added to $1 + 1 + 1 + \dots (n - 1) \text{ times}$ it becomes:

$$= 1 + 1 + 1 + \dots n \text{ times}$$

i. e. $1 \times n = n$ and time complexity is :

$$= O(n).$$
