

❖ *Algorithm*

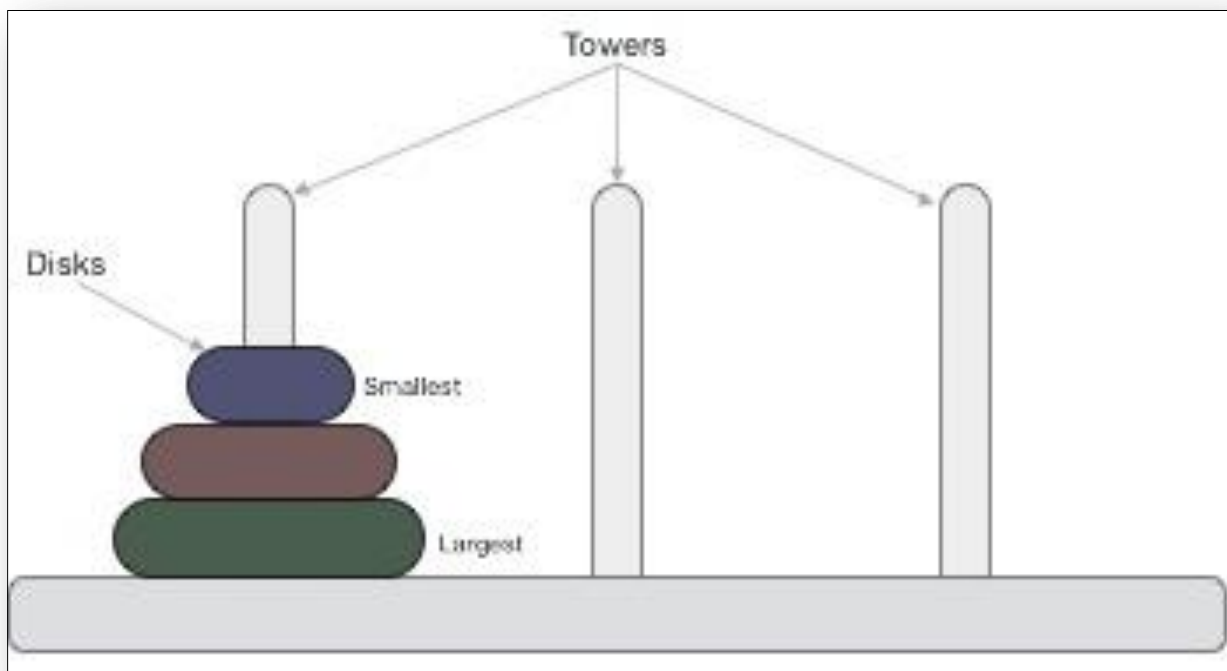
To write an algorithm for Tower of Hanoi, first we need to learn how to solve this problem with lesser amount of disks, say \rightarrow 1 or 2. We mark three towers with name, **source**, **destination** and **aux** (only to help moving the disks). If we have only one disk, then it can easily be moved from source to destination peg.

If we have 2 disks –

- First, we move the smaller (top) disk to aux peg
- Then, we move the larger (bottom) disk to destination peg.
- And finally, we move the smaller disk from aux to destination peg.
- So now, we are in a position to design an algorithm for Tower of Hanoi with more than two disks. We divide the stack of disks in two parts.
- The largest disk (n^{th} disk) is in one part and all other ($n-1$) disks are in the second part.
- Our ultimate aim is to move disk n from source to destination and then put all other ($n-1$) disks onto it. We can imagine to apply the same in a recursive way for all given set of disks.

The steps to follow are –

- **Step 1** – Move $n-1$ disks from **source** to **aux**
- **Step 2** – Move n^{th} disk from **source** to **dest**
- **Step 3** – Move $n-1$ disks from **aux** to **dest**



A recursive algorithm for Tower of Hanoi can be driven as follows –

```

START
Procedure
Hanoi(disk, source, dest, aux)

    IF disk ==1, THEN
        move disk from source to dest
    ELSE
        Hanoi(disk -1, source, aux, dest)// Step 1
        move disk from source to dest
    // Step 2
    Hanoi(disk -1, aux, dest, source)// Step 3
    END IF

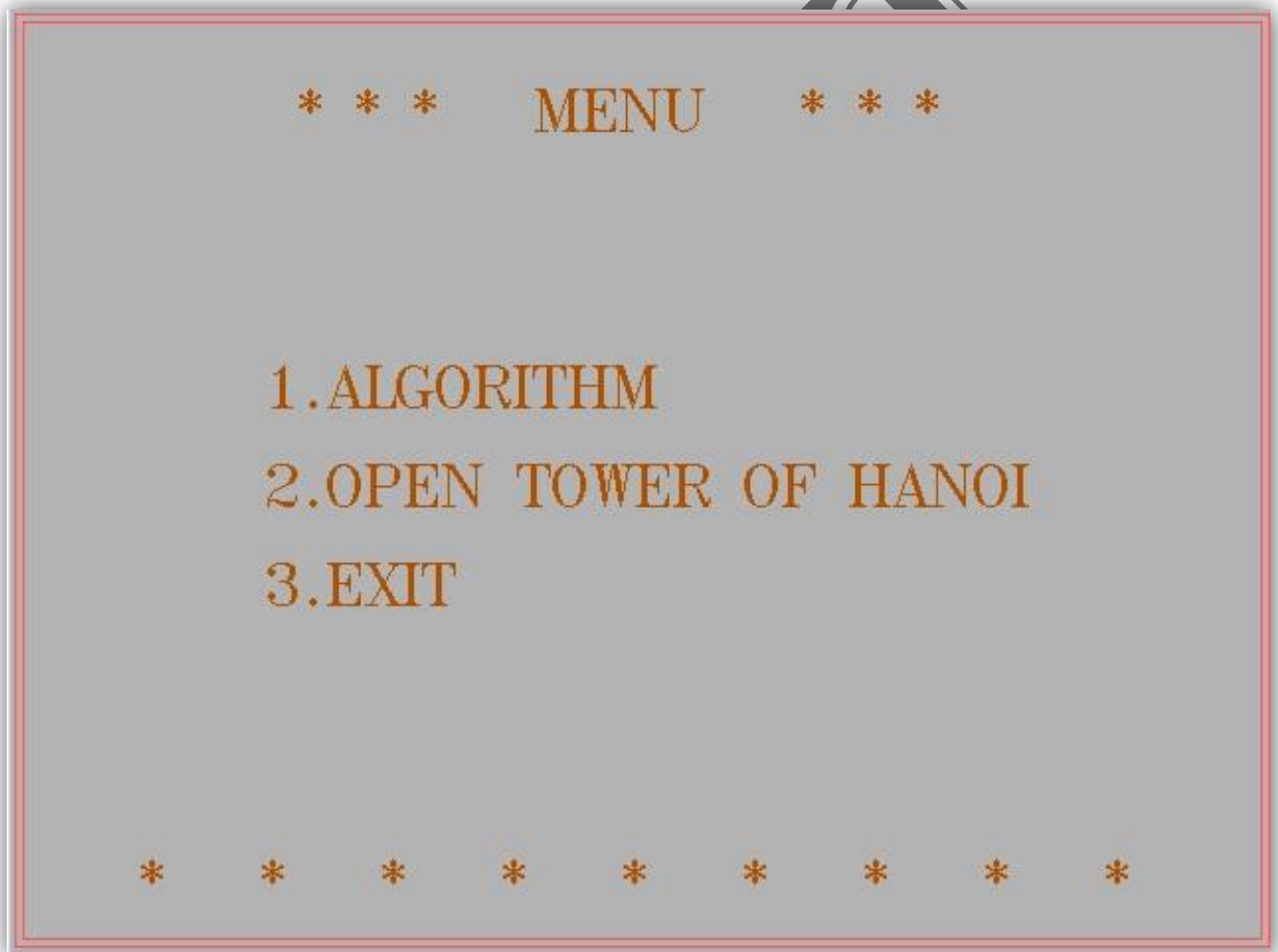
ENDProcedure
STOP
  
```

❖ *OUTPUT:*

1. Welcome page



2. Menu page



3. Selecting Algorithm page

ALGORITHM ~

TowerOfHanoi(N, A, B, C) /N is the no. of disks

/Move the top n disks from tower A to tower B

{ if(N>1) then

{ TowerOfHanoi(N-1, A, C, B);

write('move top disk from tower ', A 'to top of tower', B);

TowerOfHanoi(N-1, C, B, A);

} }

Press any key to go back..

1. *Entering No. Of Disks to be entered (MAX 10).*

(Here $n=3$)



* * * MENU * * *

1.ALGORITHM

2.OPEN TOWER OF HANOI

3.EXIT

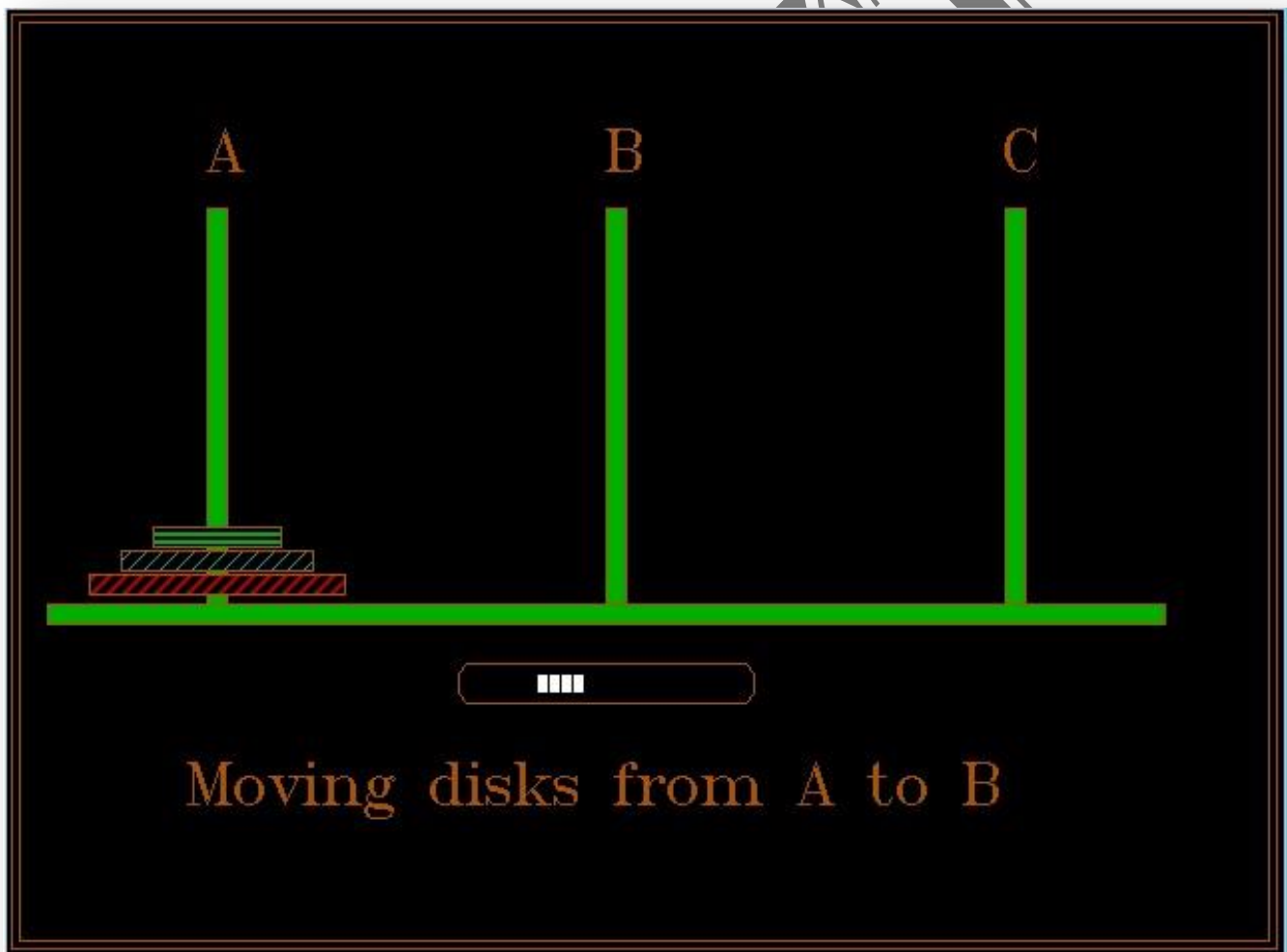
NOW, DISKS TO BE ENTERED (max 10):-

* * * * *

2. *Tower of Hanoi page*

Moving disks from A to B

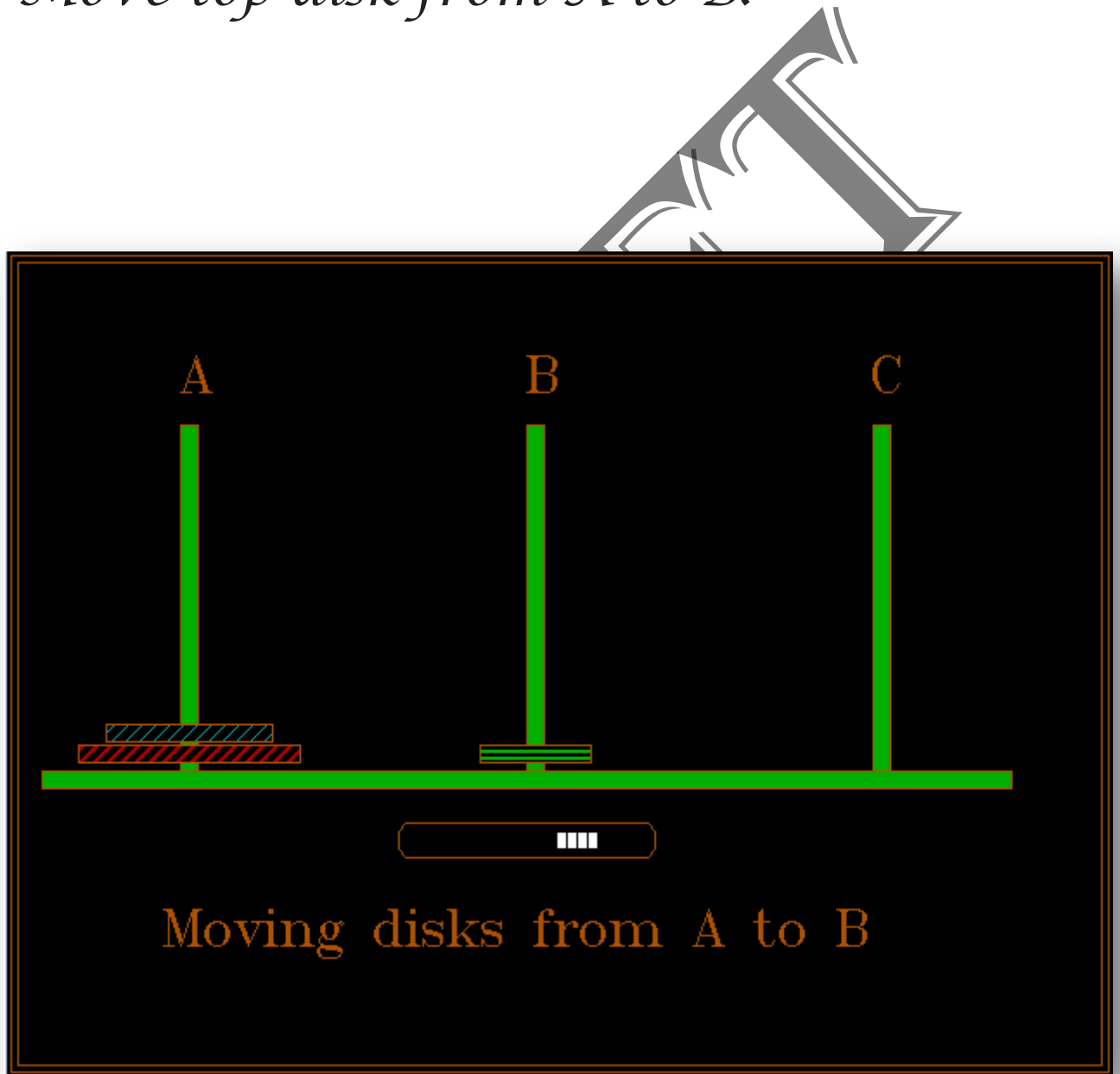
(Total Moves $2^n - 1 = 8 - 1 = 7$ moves)



- *INITIAL STATE*

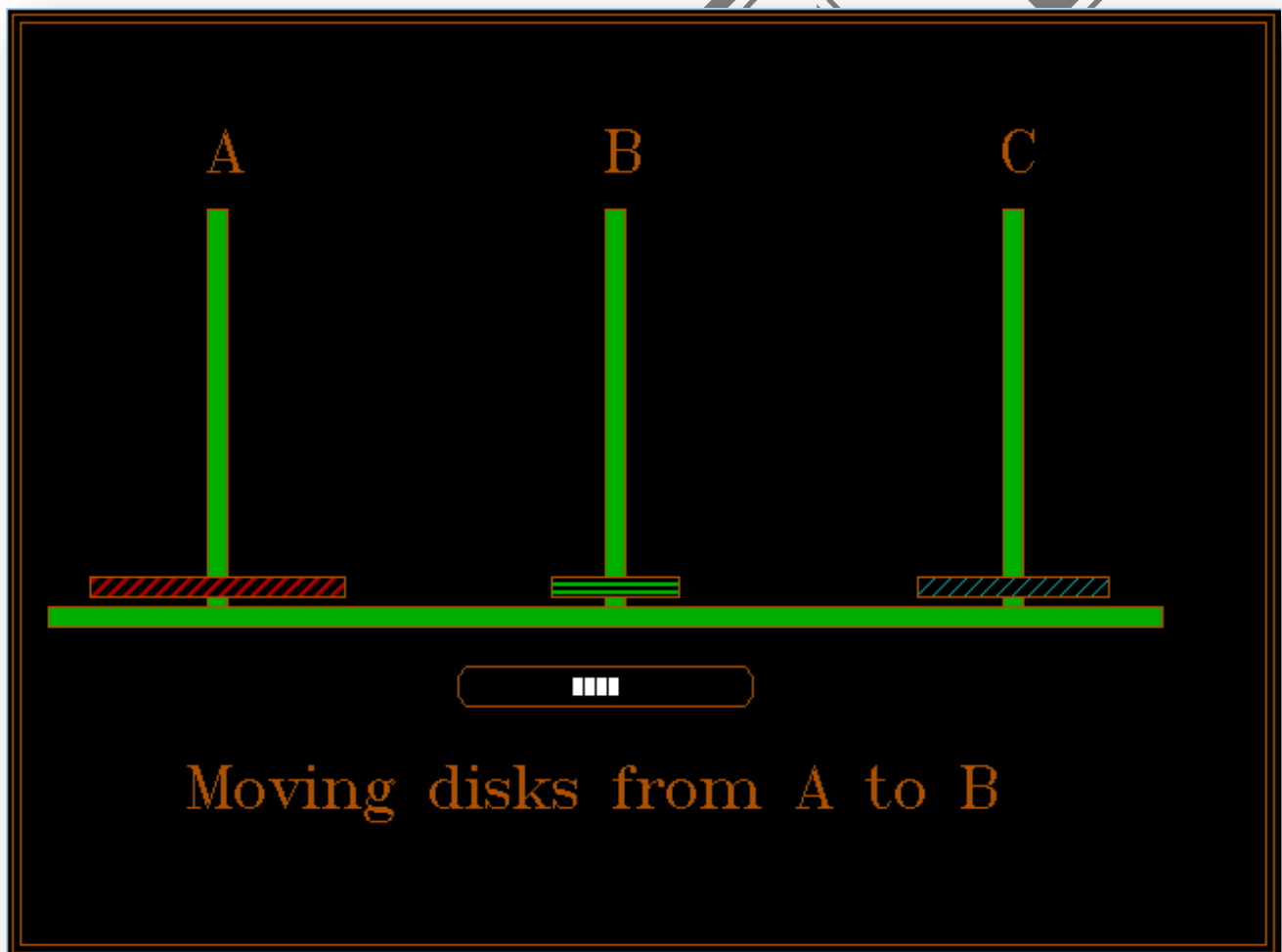
STEP 1:

Move top disk from A to B.



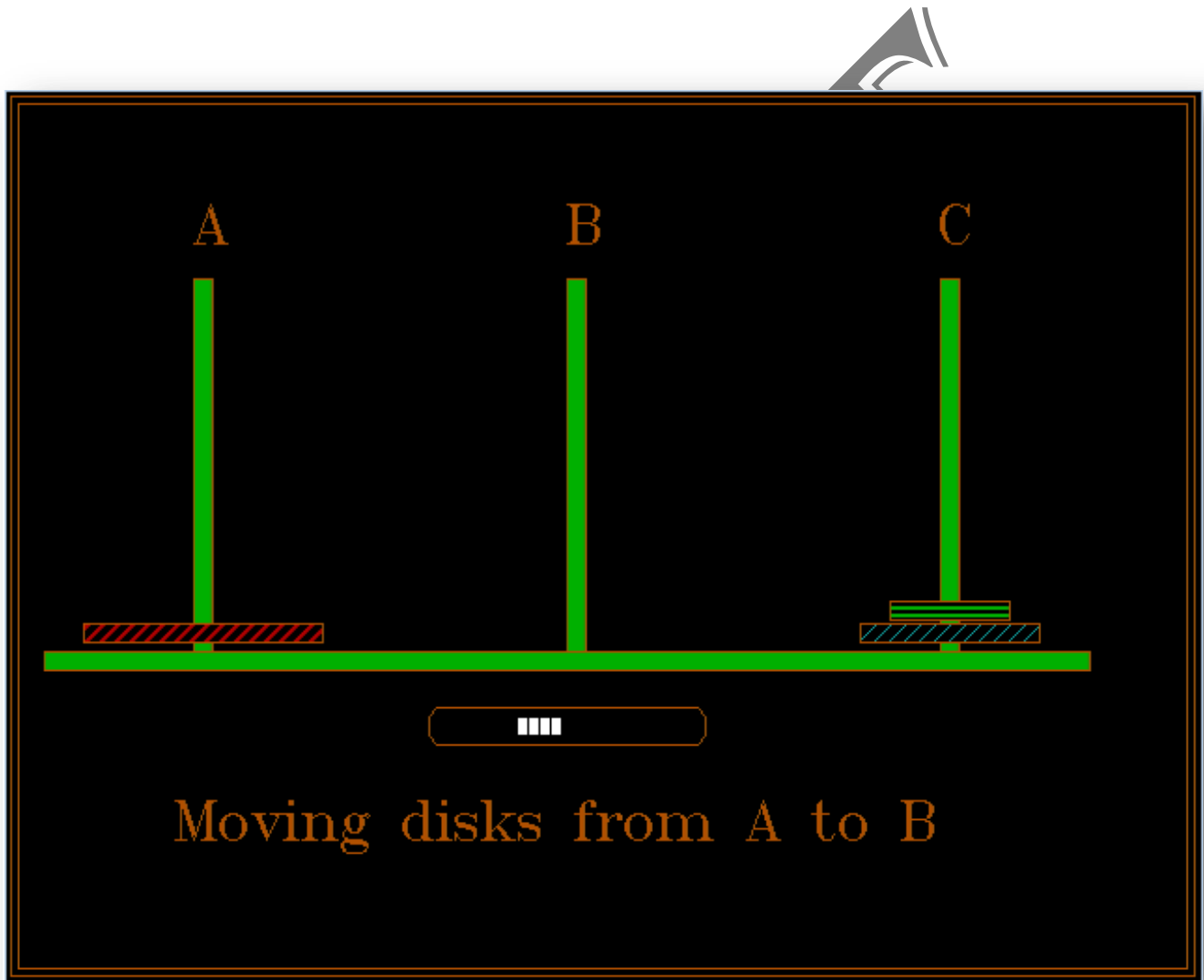
STEP 2:

Move top disk from A to C.



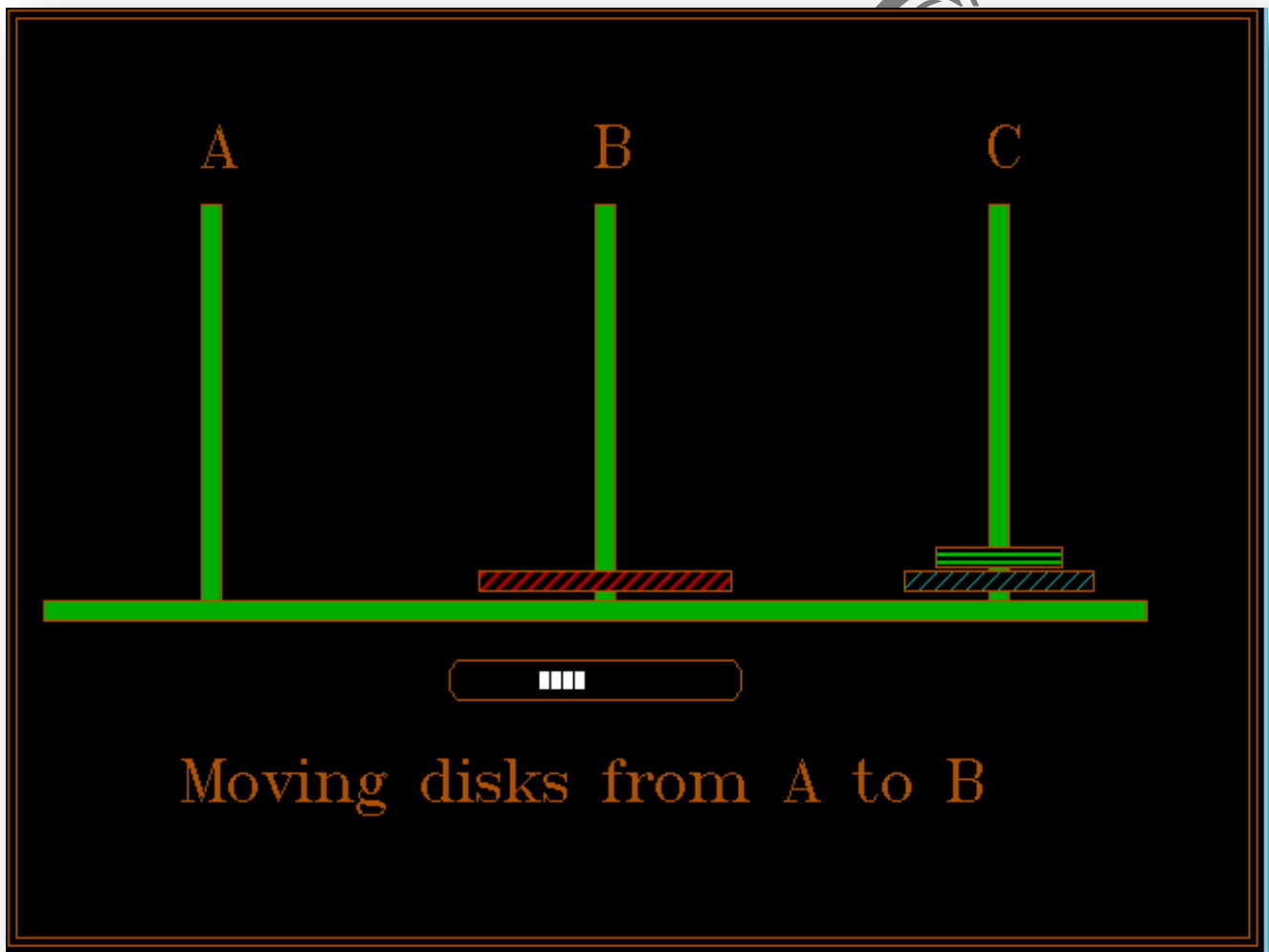
STEP 3:

Move top disk from B to C.



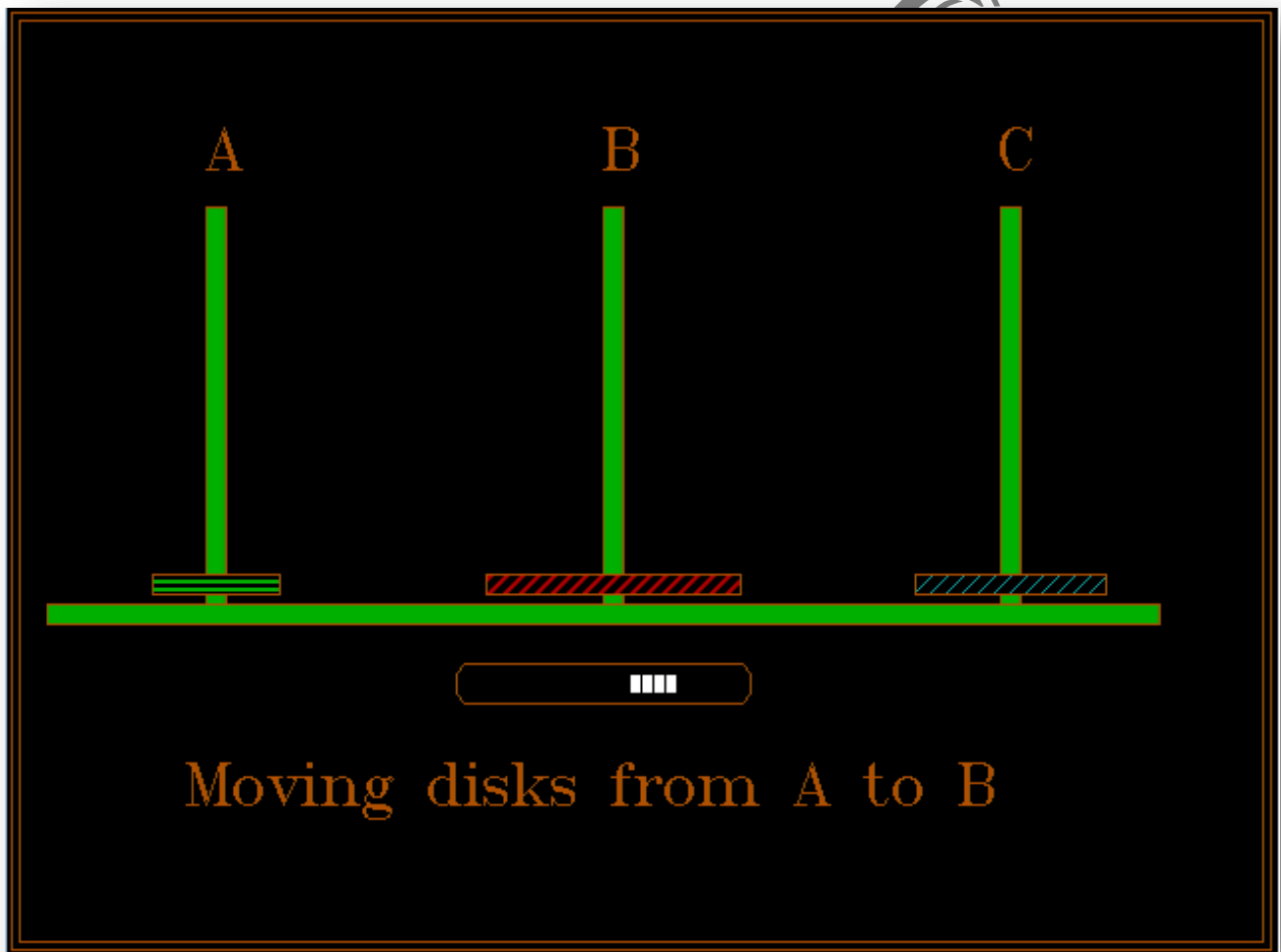
STEP 4:

Move top disk from A to B.



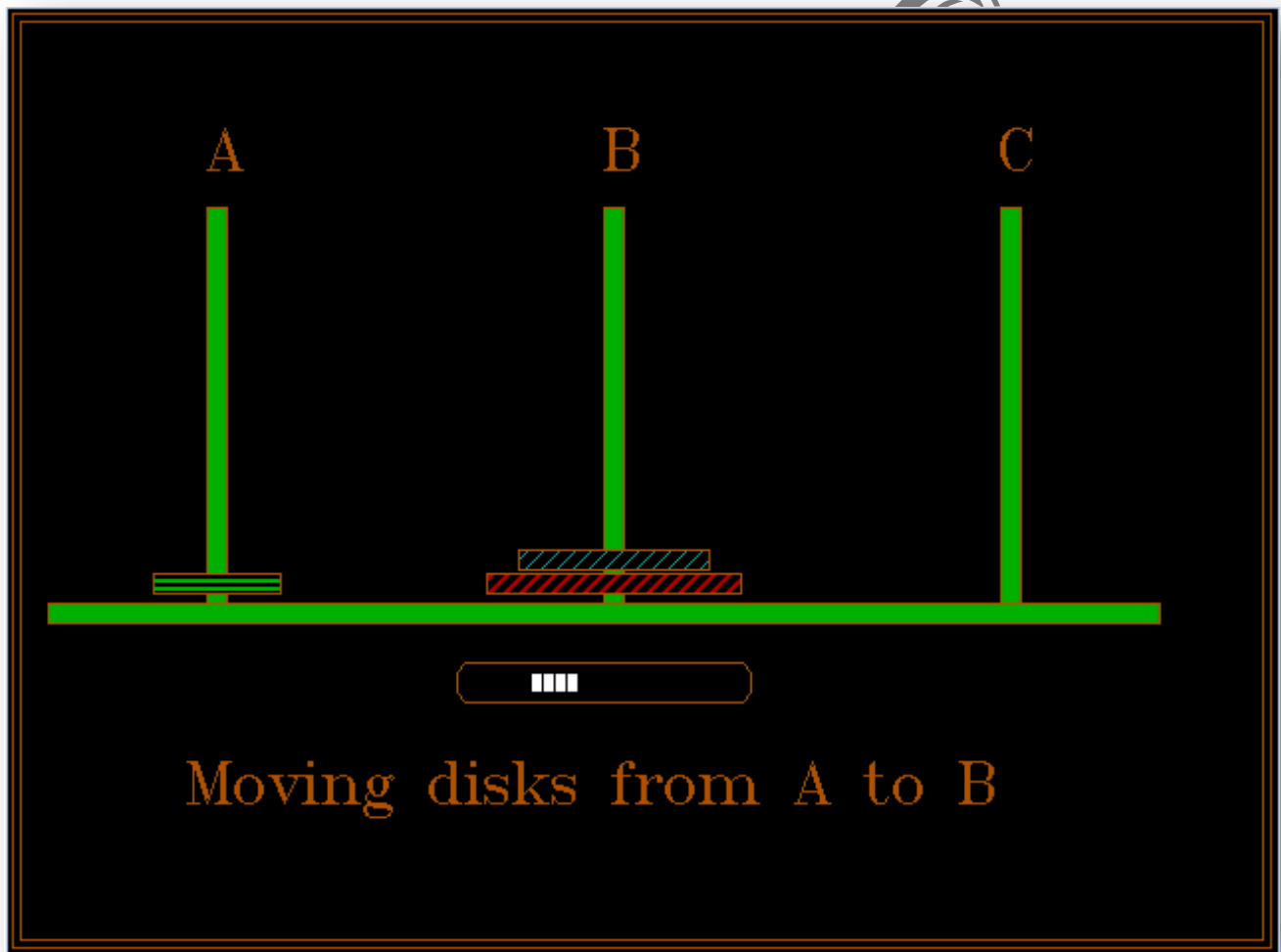
STEP 5:

Move top disk from C to A.



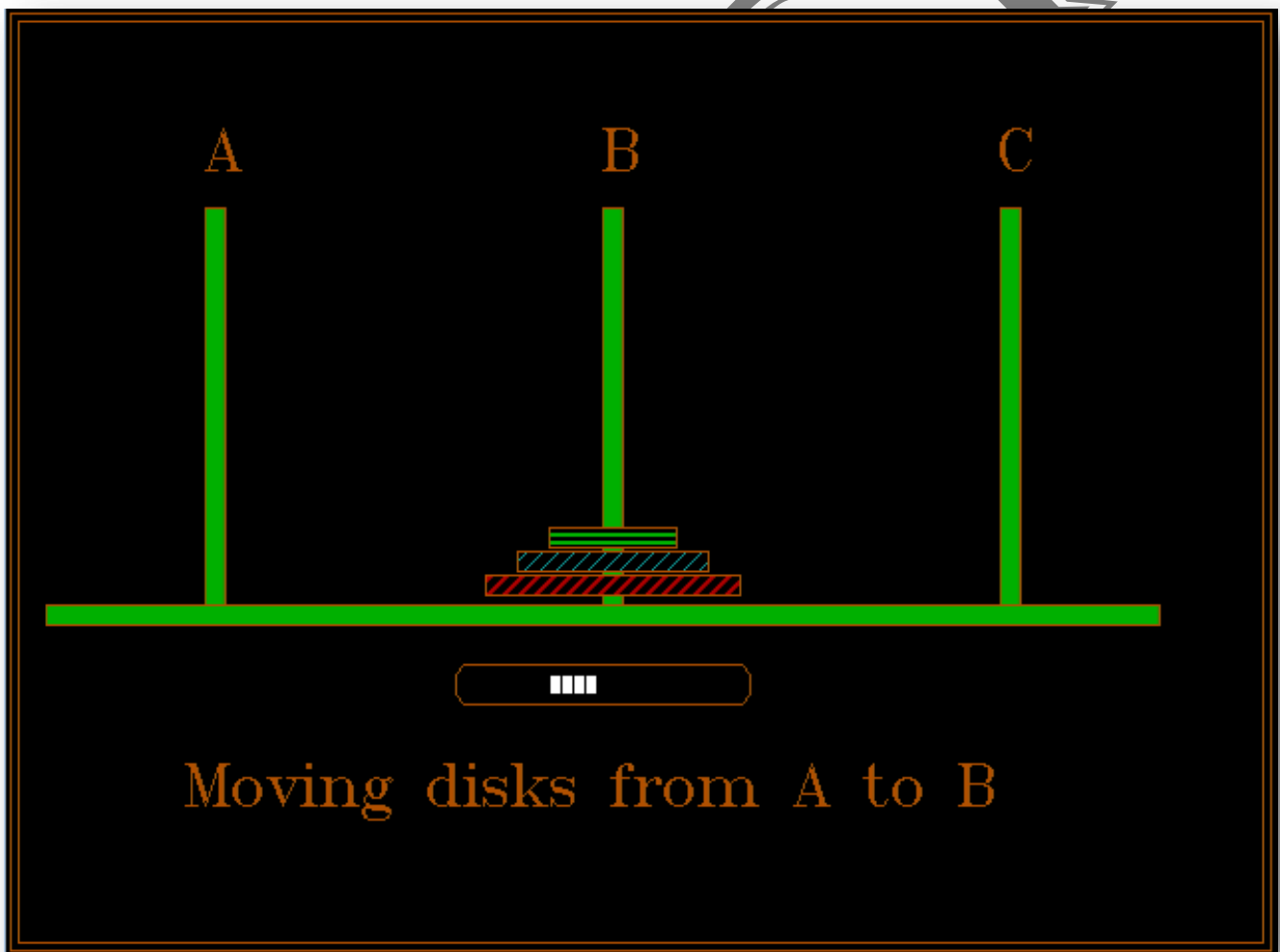
STEP 6:

Move top disk from C to B.



STEP 7:

Move top disk from A to B.



- *FINAL STATE*