incode

Business Case.

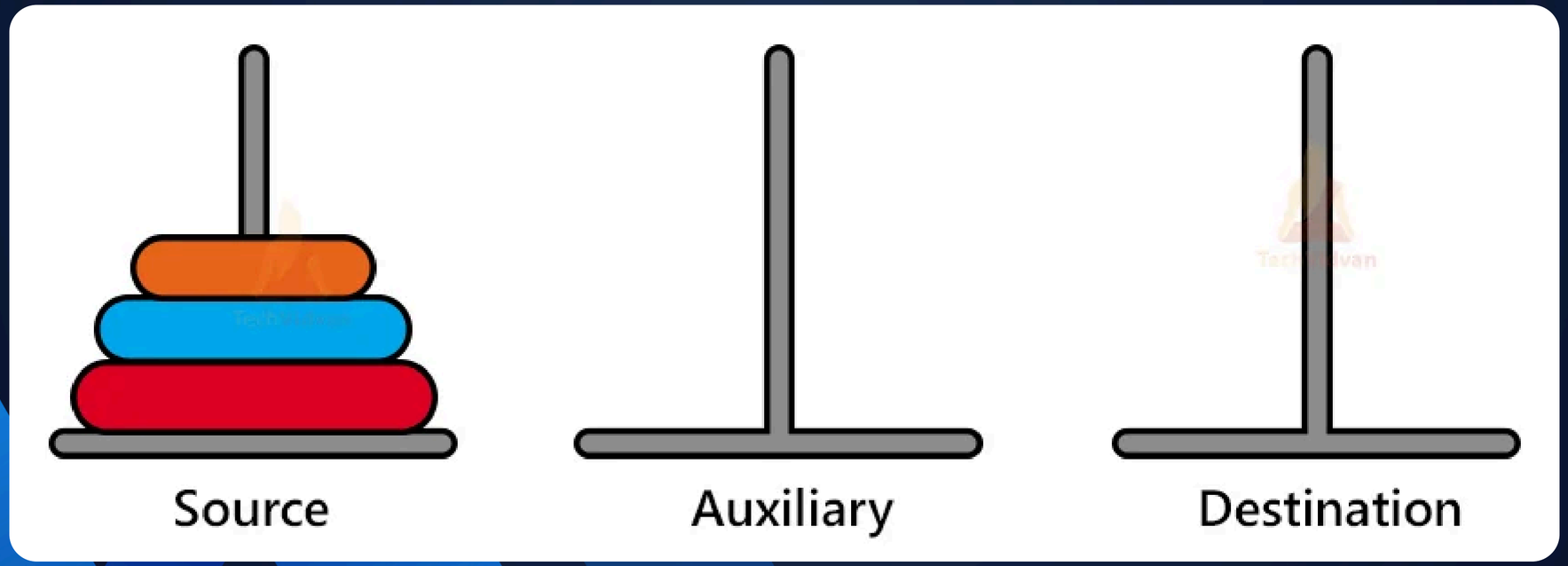
Tower of Hanoi

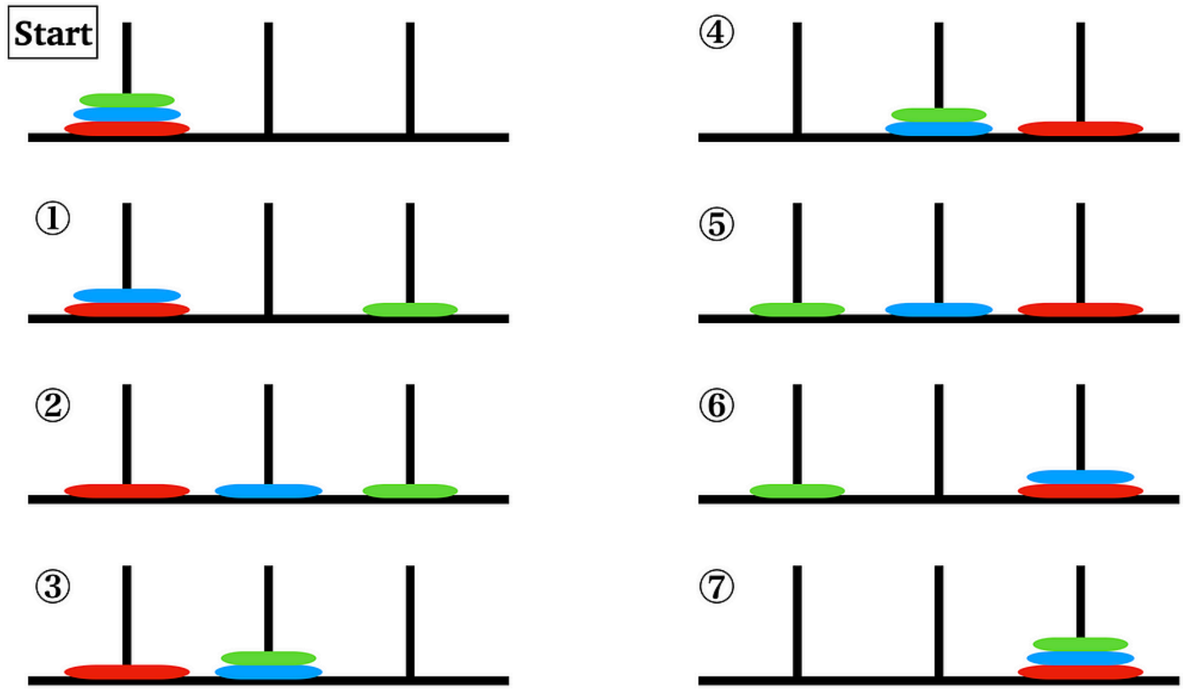
Presented by Hristina Nikolic

October 2024

incode

Tower of Hanoi



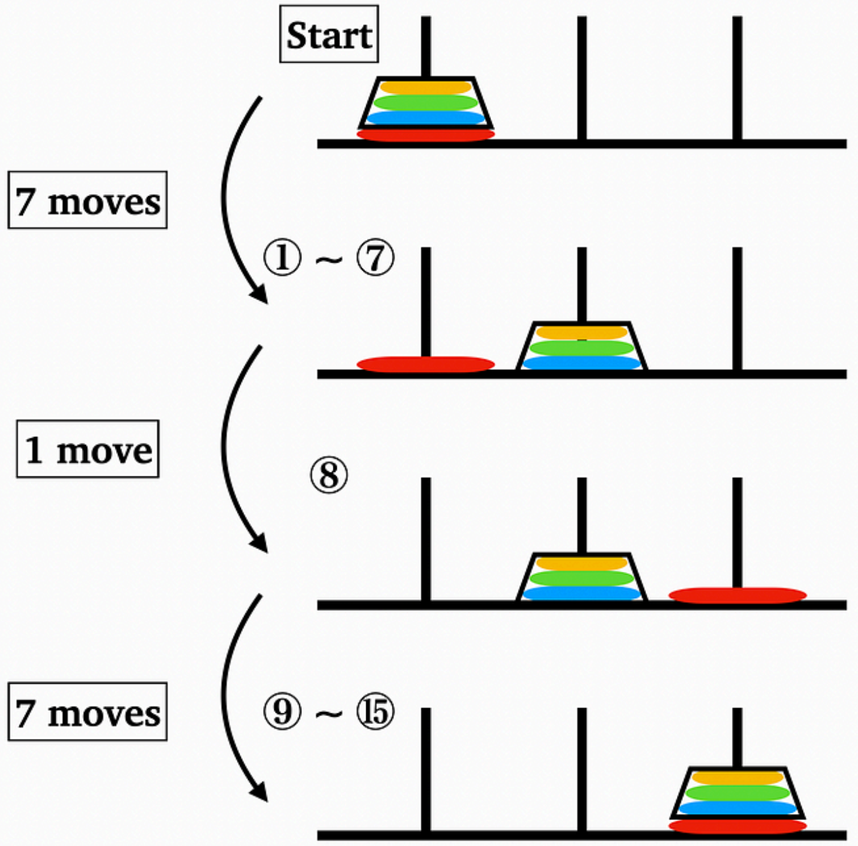

Start
①
②
③
④
⑤
⑥
⑦

incode

Recursive Problem Solving

Let f(n) be a recursive function.

1. Show f(1) works. (base case)
2. Assume f(n-1) works.
3. Show f(n) works using f(n-1).


Start
7 moves
① ~ ⑦
1 move
⑧
7 moves
⑨ ~ ⑮

incode

My solution

hristinanikpi/
tower_of_hanoi

The project was done as part of the interview process for the position of Junior Solutions Engineer at Incode Technologies.

1 Contributor 0 Issues 0 Stars 0 Forks

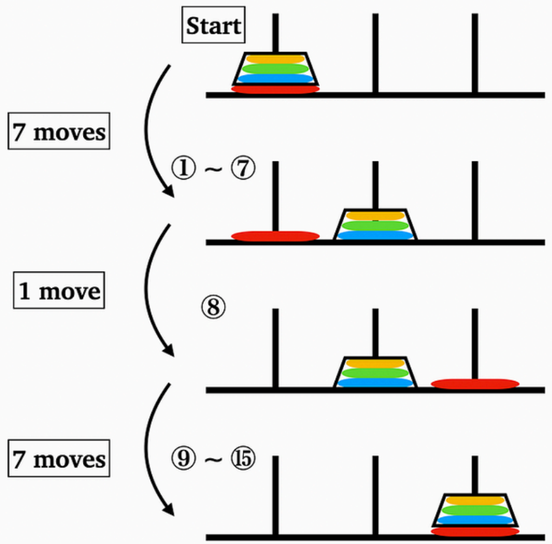
hristinanikpi/tower_of_hanoi: The project was done as part of the interview process for the position of Junior...

The project was done as part of the interview process for the position of Junior Solutions Engineer at Incode Technologies. - hristinanikpi/tower_of_hanoi

GitHub

incode

```python
def tower_of_hanoi_recursive(n, source, destination, auxiliary, moves):
    if n == 1:
        moves.append(f"Take disk 1 from rod {source} to rod {destination}.")
        return
    tower_of_hanoi_recursive(n-1, source, auxiliary, destination, moves)
    moves.append(f"Take disk {n} from rod {source} to rod {destination}.")
    tower_of_hanoi_recursive(n-1, auxiliary, destination, source, moves)
```



The Recursive Algorithm

incode

Tech Stack

Back-end

Python 3

Front-end

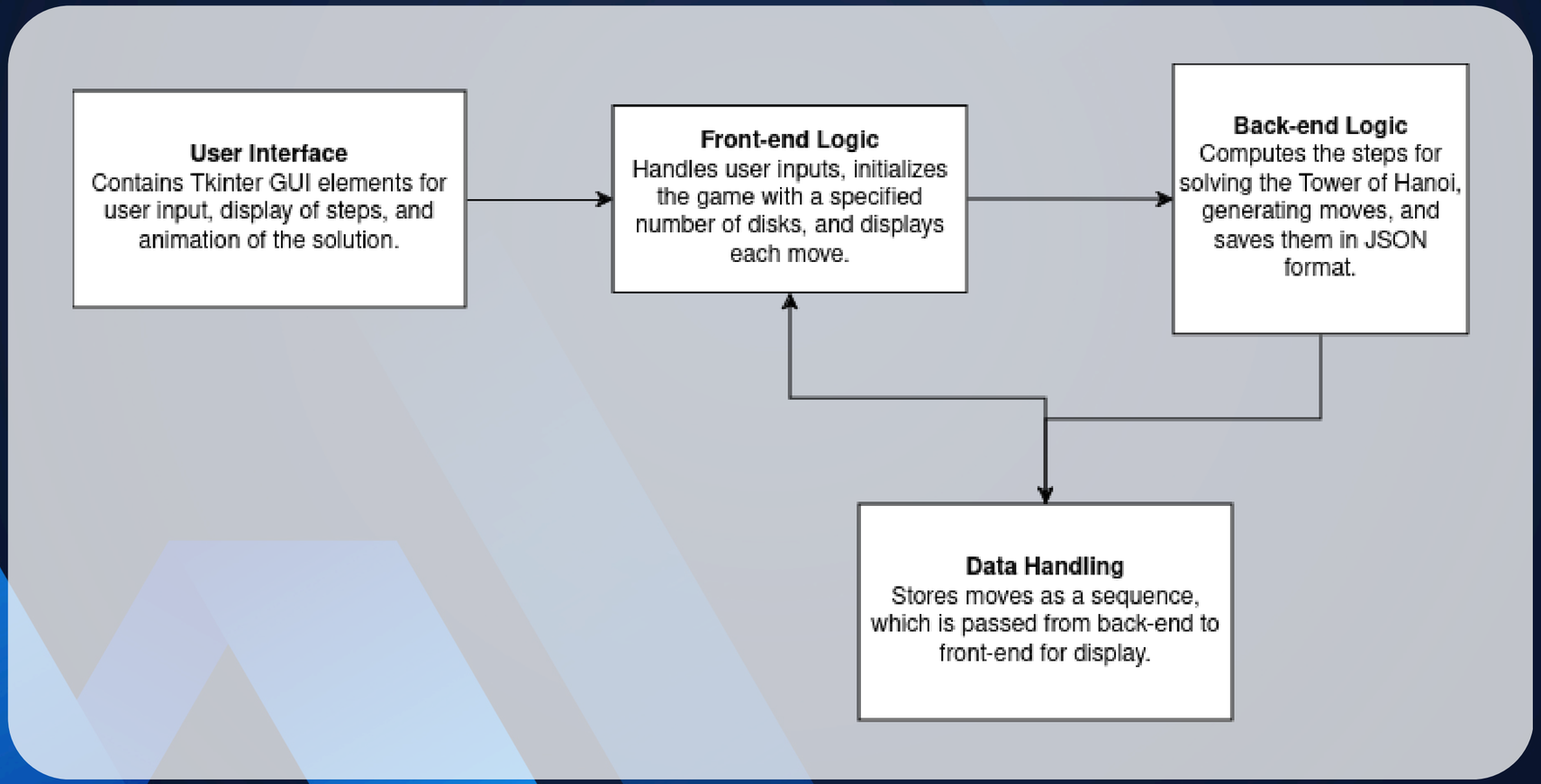
Python (Tkinter)

Data

JSON object

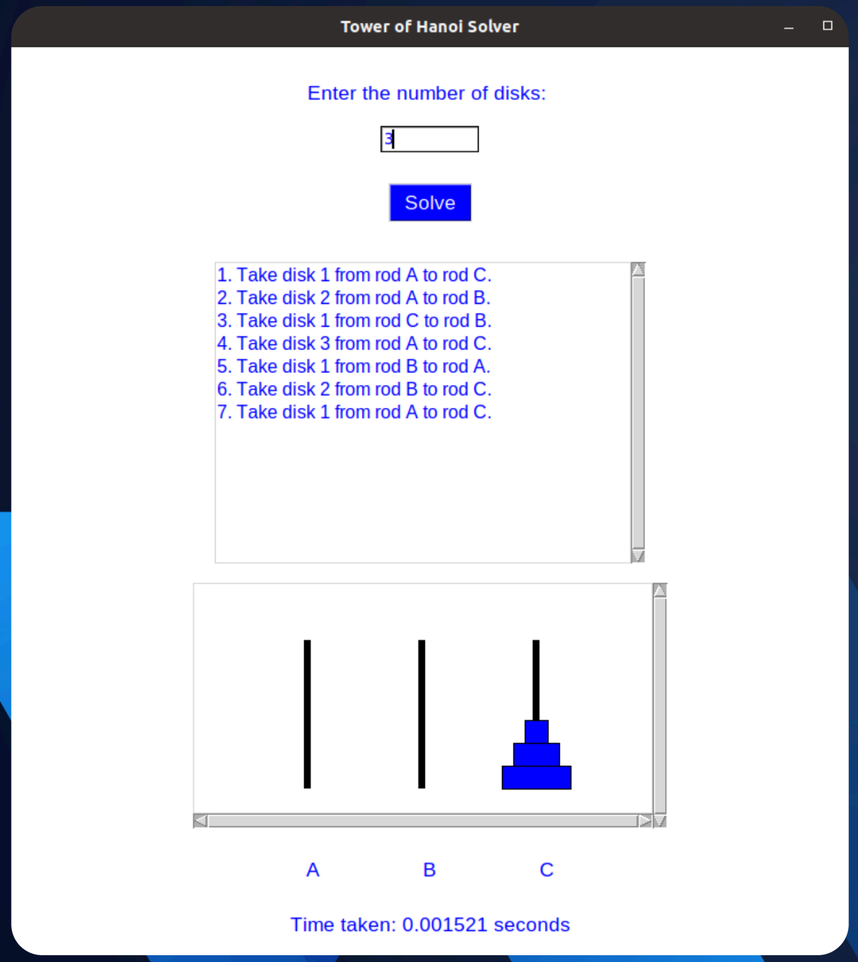
incode

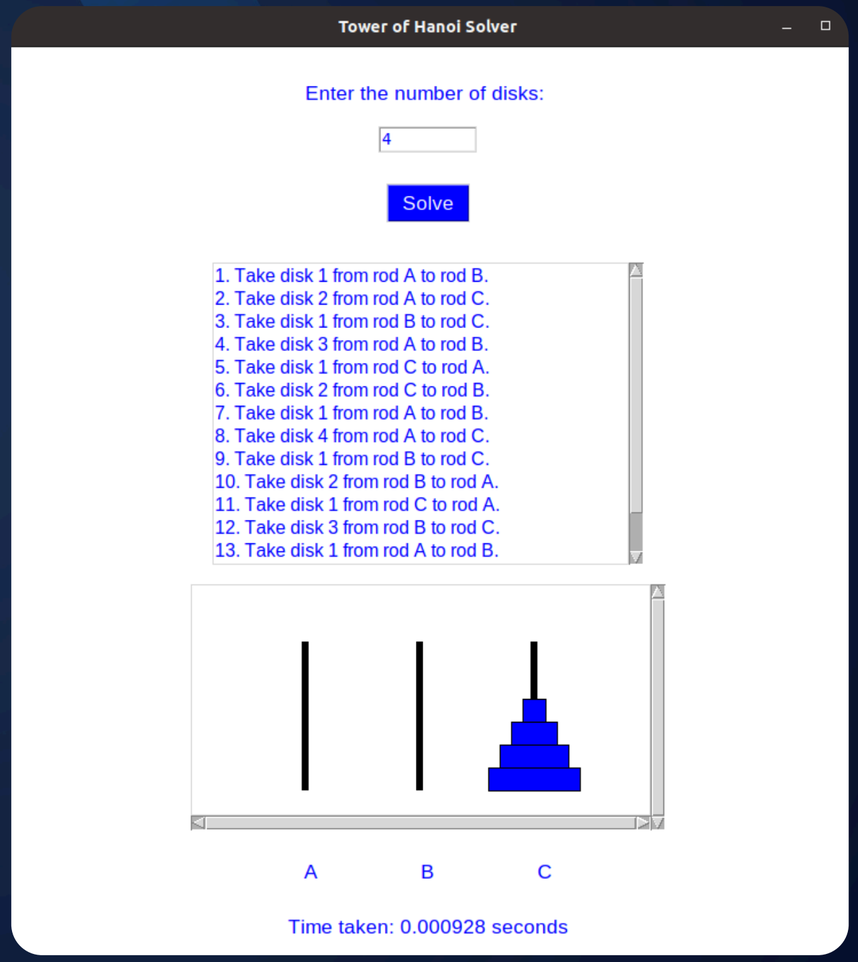
Application Diagram



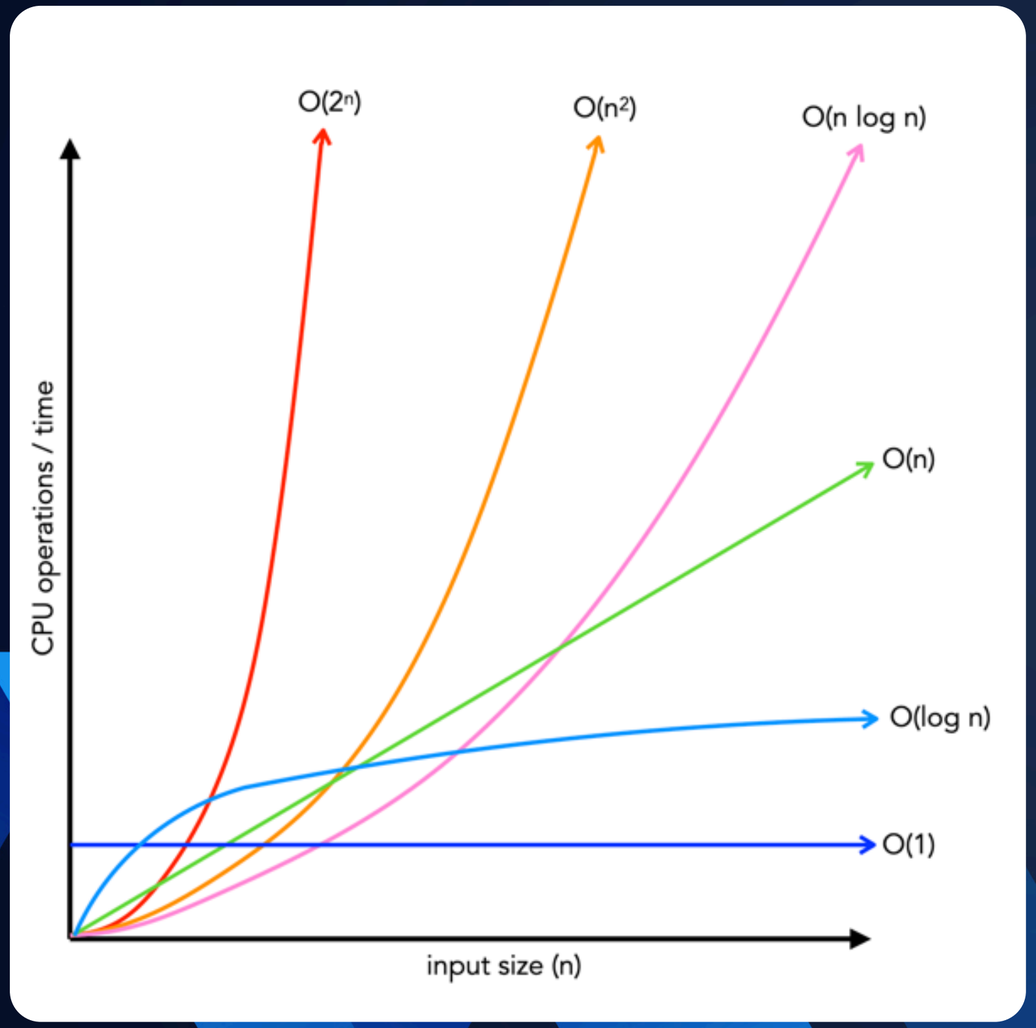
incode

Demo


Tower of Hanoi Solver
Enter the number of disks:
3
Solve
1. Take disk 1 from rod A to rod C.
2. Take disk 2 from rod A to rod B.
3. Take disk 1 from rod C to rod B.
4. Take disk 3 from rod A to rod C.
5. Take disk 1 from rod B to rod A.
6. Take disk 2 from rod B to rod C.
7. Take disk 1 from rod A to rod C.
A
B
C
Time taken: 0.001521 seconds


Tower of Hanoi Solver
Enter the number of disks:
4
Solve
1. Take disk 1 from rod A to rod B.
2. Take disk 2 from rod A to rod C.
3. Take disk 1 from rod B to rod C.
4. Take disk 3 from rod A to rod B.
5. Take disk 1 from rod C to rod A.
6. Take disk 2 from rod C to rod B.
7. Take disk 1 from rod A to rod B.
8. Take disk 4 from rod A to rod C.
9. Take disk 1 from rod B to rod C.
10. Take disk 2 from rod B to rod A.
11. Take disk 1 from rod C to rod A.
12. Take disk 3 from rod B to rod C.
13. Take disk 1 from rod A to rod B.
A
B
C
Time taken: 0.000928 seconds

incode



Time Complexity

The minimal number of moves required to solve a Tower of Hanoi puzzle is 2^n - 1, where n is the number of disks.

$$O(2^n)$$

incode

Solving Time

5 disks

	recursive	iterative
1st try	0.001874	0.002447
2nd try	0.001886	0.002070
3rd try	0.001813	0.001859
4th try	0.001815	0.002078
5th try	0.001800	0.002070

10 disks

	recursive	iterative
1st try	0.016476	0.017183
2nd try	0.017035	0.019195
3rd try	0.016347	0.017300
4th try	0.016690	0.018979
5th try	0.016434	0.022268

20 disks

	recursive	iterative
1st try	3.537602	3.672936
2nd try	3.583698	3.574592
3rd try	3.535073	3.598811
4th try	3.592360	3.672235
5th try	3.502956	3.592561

Solving Time

50 disks

while 5 disks take a manageable time, 50 disks would take 1 125 899 906 842 624 moves, which is infeasible to compute in a reasonable time.

Future Improvements

- Enhanced Visuals
- Web and Mobile Compatibility
- Alternative Algorithms
- Statistics Tracking

incode

THANK YOU