

# **Tower of Hanoi Programmer Manual**

## **Description**

The project was aimed to solve the tower of Hanoi problem to move disks from one pole to another pole. During the solution input prompt will be displayed to user, upon entering required inputs, inputs will be validated for correct values and finally the problem will be solved using recursive function calling.

## **Functions Used**

### **Greet**

This function was used to display the greeting message in the console screen.

### **Input**

This function was used to get the input from user in the variables num\_disk, start\_pole and end\_pole. Before taking each input a prompt message is displayed to user for input to guide the user which value has to be entered.

### **Validate**

This function was used to validate the input. No of disk should be in the range from 1 to 7 while start pole and end pole should be in the range of 1 to 3. Also, start pole and end pole can't be the same. If any validation fails a message will be displayed on console screen telling that the values entered are invalid.

### **Hanoi**

This is the main function of the project which will call itself recursively and show the sequence of steps to solve the problem. It has 3 parameters num\_disk, start\_pole and end\_pole. In this function we check first if number of disk is 1, if yes then we then we show the message to move the disk from start pole to end pole, if not then firstly we call the Hanoi function recursively by altering the parameters as

- num\_disk = num\_disk – 1
- start\_pole = start\_pole
- end\_pole = 6 - start\_pole - end\_pole

After that we show the message to move the disk from start pole to end pole and call the same function recursively once again but this time with other parameters modification

- num\_disk = num\_disk – 1
- start\_pole = 6 - start\_pole - end\_pole
- end\_pole = end\_pole

After each recursive call, we pop the parameters from stack to clean it because we had pushed these parameters in stack for the function.

## Algorithm Used

```
Procedure Hanoi(n, r, s)
if (n == 1) {
    print("move disk ",n, " from pole ",r, " to pole ", s)
} else {
    H(n-1, r, 6-r-s) // the pole not used = 6-r-s since 6= 1+2+3
    print ("move disk ",n, " from pole ", r, " to pole ", s)
    H(n-1, 6-r-s, s)
}
return // end of procedure
```

## Instructions to run

1. Copy Source files
  - a. Open C: drive and navigate to **TASM** folder.
  - b. There will be another **tasm** folder, navigate to that folder
  - c. Create a new folder named "**CS221**" and navigate to this new folder
  - d. Create a folder named "**P2**" and open it.
  - e. Now copy the source files "**Amin\_P2.asm**" and "**iofar.lib**" here.
2. Prepare environment
  - a. Open **DOSBOX** application
  - b. At the prompt widows type "**cd tasm**"
  - c. Now you will see something like "**C:\TASM>**", type "**cd CS221**" to move to next folder.
  - d. Next, go to the P2 folder where the actual source files are placed by typing "**cd P2**" command.
  - e. We are now in the folder of source files and ready to execute
3. Compiling and running project
  - a. Now run the following commands
    - i. `tasm /zi/la/c/ml Amin_P2.asm`
    - ii. `tlink /v/s/m/3 Amin_P2,,,c:iofar`
  - b. Our project has been compiled and an executable file named "**Amin\_P2.exe**" has been created. Type "**Amin\_P2.exe**" command to run it.
  - c. Enter number of disks, start pole and end pole and a sequence of steps will be displayed to move all the disks from start pole to end pole without breaking the rules.
4. End.