**CODING ASSIGNMENT 9.1 SOLUTION**

**Index.html:**

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<title>Tower of Hanoi Step by Step</title>

<style>

body {

background-color: teal;

font-family: Arial, sans-serif;

padding: 10px;

}

.controls {

margin-bottom: 20px;

}

.rod {

width: 20px;

height: 300px;

background-color: lightgray;

margin: 0 120px;

position: relative;

display: inline-block;

vertical-align: bottom;

}

.disk {

position: absolute;

height: 25px;

background-color: crimson;

text-align: center;

color: black;

font-size: 14px;

line-height: 25px;

border-radius: 3px;

}

</style>

</head>

<body>

<div class="controls">

<input type="number" id="diskCount" value="5" min="1" max="8">

<button onclick="makeMove()">Make Move</button>

</div>

<div id="rod1" class="rod"></div>

<div id="rod2" class="rod"></div>

<div id="rod3" class="rod"></div>

<script>

let numDisks;

const rods = {

A: document.getElementById('rod1'),

B: document.getElementById('rod2'),

C: document.getElementById('rod3')

};

let state = { A: [], B: [], C: [] };

let moves = [];

let moveIndex = 0;

function initGame() {

for (let r of ['A','B','C']) {

rods[r].innerHTML = '';

state[r] = [];

}

moves = [];

moveIndex = 0;

numDisks = parseInt(document.getElementById('diskCount').value);

for (let i = numDisks; i >= 1; i--) {

let disk = document.createElement('div');

disk.className = 'disk';

disk.style.width = (60 + i \* 20) + 'px';

disk.textContent = i;

rods.A.appendChild(disk);

state.A.push(disk);

positionDisks('A');

}

// Pre-calculate all moves

hanoi(numDisks, 'A', 'C', 'B');

}

function positionDisks(rod) {

let disks = state[rod];

for (let i = 0; i < disks.length; i++) {

let d = disks[i];

d.style.bottom = (i \* 26) + 'px';

d.style.left = (-d.offsetWidth / 2 + 10) + 'px';

}

}

function hanoi(n, from, to, aux) {

if (n === 1) {

moves.push([from, to]);

} else {

hanoi(n-1, from, aux, to);

moves.push([from, to]);

hanoi(n-1, aux, to, from);

}

}

function makeMove() {

if (moveIndex >= moves.length) {

alert("Tower is Finished");

return;

}

let [from, to] = moves[moveIndex];

let disk = state[from].pop();

state[to].push(disk);

rods[to].appendChild(disk);

positionDisks(from);

positionDisks(to);

moveIndex++;

if (moveIndex === moves.length) {

alert("Tower is Finished");

}

}

window.onload = initGame;

</script>

</body>

</html>