1.

#include <iostream>

#include <stack>

using namespace std;

struct Task {

int n;

char src, aux, dest;

int state; // 0 = not processed yet, 1 = left done (ready to move biggest disk)

};

void hanoi\_iterative(int n, char src, char aux, char dest) {

stack<Task> st;

st.push({n, src, aux, dest, 0});

while (!st.empty()) {

Task t = st.top();

st.pop();

if (t.n == 1) {

cout << "Move disk 1 from " << t.src << " to " << t.dest << "\n";

} else {

if (t.state == 0) {

// Step 3 & 2 to be handled later, push current task with state=1

st.push({t.n, t.src, t.aux, t.dest, 1});

// Step 1: Move n-1 disks from src to aux

st.push({t.n - 1, t.src, t.dest, t.aux, 0});

}

else if (t.state == 1) {

// Step 2: Move largest disk

cout << "Move disk " << t.n << " from " << t.src << " to " << t.dest << "\n";

// Step 3: Move n-1 disks from aux to dest

st.push({t.n - 1, t.aux, t.src, t.dest, 0});

}

}

}

}

int main() {

int n;

cout << "Enter number of disks: ";

cin >> n;

cout << "The sequence of moves is:\n";

hanoi\_iterative(n, 'A', 'B', 'C');

return 0;

}

2 . stack

#include <iostream>

using namespace std;

#define MAX 100

class Stack {

int arr[MAX];

int top;

public:

Stack() { top = -1; }

// Push element

void push(int x) {

if (top == MAX - 1) {

cout << "Stack Overflow\n";

} else {

arr[++top] = x;

cout << x << " pushed into stack\n";

}

}

// Pop element

void pop() {

if (top == -1) {

cout << "Stack Underflow\n";

} else {

cout << arr[top--] << " popped from stack\n";

}

}

// Peek element (top of stack)

void peek() {

if (top == -1) {

cout << "Stack is empty\n";

} else {

cout << "Top element is: " << arr[top] << "\n";

}

}

// Display all elements

void display() {

if (top == -1) {

cout << "Stack is empty\n";

} else {

cout << "Stack elements: ";

for (int i = top; i >= 0; i--) {

cout << arr[i] << " ";

}

cout << "\n";

}

}

};

int main() {

Stack st;

int choice, value;

while (true) {

cout << "\n--- Stack Menu ---\n";

cout << "1. Push\n";

cout << "2. Pop\n";

cout << "3. Peek\n";

cout << "4. Display\n";

cout << "5. Exit\n";

cout << "Enter your choice: ";

cin >> choice;

switch (choice) {

case 1:

cout << "Enter value to push: ";

cin >> value;

st.push(value);

break;

case 2:

st.pop();

break;

case 3:

st.peek();

break;

case 4:

st.display();

break;

case 5:

cout << "Exiting...\n";

return 0;

default:

cout << "Invalid choice! Try again.\n";

}

}

}