#include<iostream>

#include<queue>

using namespace std;

class Solution {

public:

queue<int> q1;

queue<int> q2;

void push(int x) {

q2.push(x);

while(!q1.empty()) {

q2.push(q1.front());

q1.pop();

}

q1.push(x);

while(!q2.empty()) {

q1.push(q2.front());

q2.pop();

}

}

void pop() {

if(q1.empty()) {

cout<<"Stack is EMPTY!!"<<endl;

return;

}

cout<<"Popped element: "<<q1.front()<<endl;

q1.pop();

}

int top() {

if(q1.empty()) {

cout<<"Stack is EMPTY!!"<<endl;

return -1;

}

return q1.front();

}

bool empty() {

return q1.empty();

}

};

int main() {

Solution s;

cout << "=== Stack using 2 Queues ===" << endl;

s.push(2);

s.push(4);

s.push(6);

s.push(8);

cout << "Top element: " << s.top() << endl;

s.pop();

cout << "Top element after pop: " << s.top() << endl;

s.pop();

s.pop();

s.pop();

s.pop();

if(s.empty()) {

cout<<"Stack is empty"<<endl;

}

else {

cout<<"Stack is not empty"<<endl;

}

return 0;

}

