```java
public class Stack<T> {
    private int size;
    private Queue<T> q1 = new Queue<>();
    private Queue<T> q2 = new Queue<>();

    // default constructor
    public Stack(){};

    // returns true if the stack is empty
    public boolean isEmpty(){return this.size == 0;}

    // pushes element onto stack
    public void push(T element){
        // System.out.println("From stack(push): pushing: " + element);
        while(!q1.isEmpty()){
            q2.enqueue(q1.dequeue());
        }
        q1.enqueue(element);

        while(!q2.isEmpty()){
            q1.enqueue(q2.dequeue());
        }

        if(q1.isEmpty()){
            q1.enqueue(element);
        }
        this.size += 1;
    }

    // pop an element off the stack
    public T pop(){
        T element = q1.dequeue();
        this.size -= 1;
        // System.out.println("From stack(push): popping: " + element);
        return element;
    }

    // return the top element of the stack without popping it
    public T getTop(){
        if(size > 0){
            return q1.getFirst();
        }
        return null;
    }
}
```