public Stack reverse() throws UnderflowException, OverflowException{

Object[] backup = new Object[s.length];

Stack reversed = new Stack(s.length);

Object elem;

int top = this.top;

int i = -1;

while(!isEmpty()){

elem = pop();

backup[++i] = elem;

reversed.push(elem);

}

s = backup;

this.top = top;

return reversed;

}

public Queue makeQueueFromStack(Stack s) throws UnderflowException, OverflowException {

Stack stack = s;

int stackSize = 0;

while(!(stack.isEmpty())){

stack.pop();

stackSize++;

}

Queue q = new Queue(stackSize);

stack = s;

stack.reverse();

while(!(stack.isEmpty())){

q.enqueue(stack.pop());

}

return q;

}