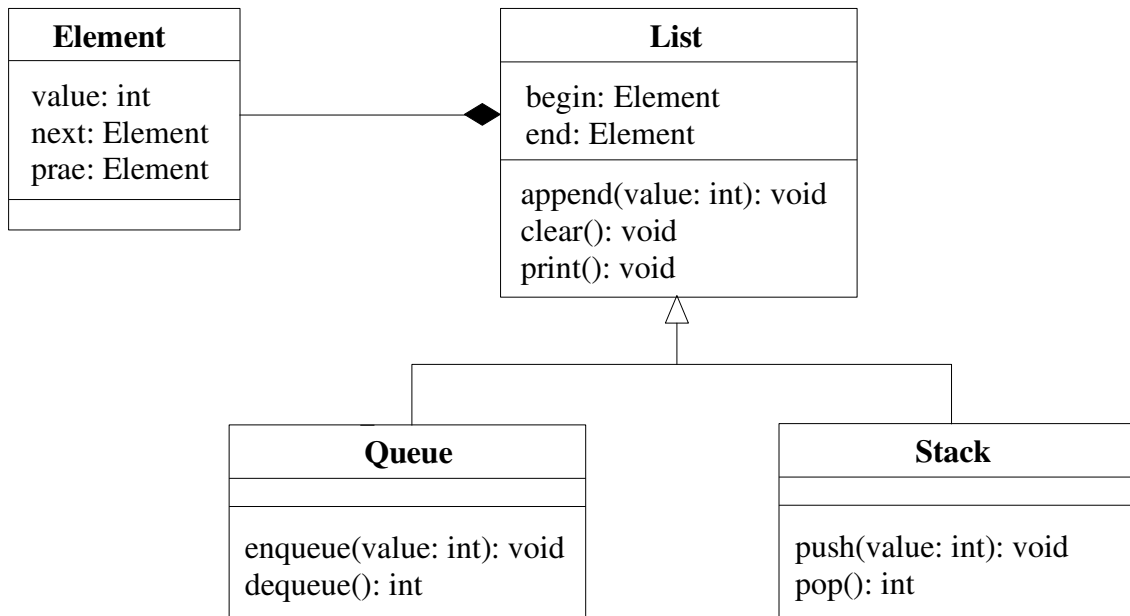


### Aufgabe 8

Implementieren Sie das folgende UML-Klassendiagramm:



*List* ist eine doppelt verkettete Liste von int-Werten als Basis für die Klassen *Queue* und *Stack*.

*append()* fügt einen int-Wert *value* an das Ende einer Liste.

*clear()* löscht die Liste (kein physisches Löschen).

*print()* gibt den Listeninhalt auf den Bildschirm aus.

*Queue* ist eine Liste vom Typ FIFO.

*enqueue()* fügt einen int-Wert dem *Queue* hinzu.

*dequeue()* liefert den zuerst hinzugefügten Wert zurück und entfernt diesen aus dem *Queue*

*Stack* ist eine Liste vom Typ LIFO.

*push()* fügt einen int-Wert dem *Stack* hinzu.

*pop()* liefert den zuletzt hinzugefügten Wert zurück und entfernt diesen aus dem *Stack*.

Bei dem Versuch, einen int-Wert aus einer leeren Liste zu entfernen, soll eine eigene Ausnahme *QueueEmptyException* bzw. *StackEmptyException* geworfen werden.

Die Ausgaben des folgenden Programms dienen der Prüfung der Funktionalität der Klassen:

```

public class Programm {
    public static void main(String[] args) {
        Queue myqueue = new Queue();
    }
}
    
```

```

try
{
    myqueue.enqueue(4);
    myqueue.enqueue(7);
    myqueue.enqueue(3);
    myqueue.print(); // 4 7 3
    System.out.println(myqueue.dequeue()+" entnommen"); // 4 entnommen
    myqueue.enqueue(8);
    myqueue.enqueue(2);
    myqueue.print(); // 7 3 8 2
    myqueue.clear();
    System.out.println(myqueue.dequeue()+" entnommen"); // FEHLER: Queue leer
}
catch(QueueEmptyException e)
{
    System.out.println("FEHLER: "+e.getMessage());
}

Stack mystack = new Stack();
try {
    mystack.push(7);
    mystack.push(3);
    mystack.print(); // 7 3
    System.out.println(mystack.pop()+" entnommen"); // 3 entnommen
    mystack.push(8);
    mystack.push(4);
    mystack.print(); // 7 8 4
    System.out.println(mystack.pop()+" entnommen"); // 4 entnommen
    System.out.println(mystack.pop()+" entnommen"); // 8 entnommen
    System.out.println(mystack.pop()+" entnommen"); // 7 entnommen
    System.out.println(mystack.pop()+" entnommen"); // FEHLER: Stack leer
}
catch(StackEmptyException e)
{
    System.out.println("FEHLER: "+e.getMessage());
}
}
}

```