

```

// DO NOT MODIFY THIS FILE
// It was generated by Hyper/J(tm) for debugging purposes only
// To modify the program, edit the source files noted in comments below

package container;

// Stub code generated by Hyper/J(tm). Use <step into> to reach actual methods.
/**
 * Fila.java
 * Class para uma Fila
 * 18 julho de 2003
 * SCE - PGCC - UFSC
 * @author Augusto César
 */

//package container;

public class Fila{

    private Node front, end;          // referencia para os nós do inicio e final
    private int size;                 // tamanho da pilha

    public Fila(){
        size = 0;
        front = null;
        end = null;
    }

    public int size(){
        return size;
    }

    public boolean isEmpty(){
        if (front==null)
            return true;
        return false;
    }

    public Object front(){
        if(isEmpty())
            throw new RuntimeException("Fila está vazia.");
        return front.getElement();
    }

    public void enqueue( Object elem ){
        Node no = new Node();
        no.setElement(elem);
        no.setNext(null);           //nó será o novo nó do final
        if (size == 0)
            front = no;           // caso de uma lista previamente vazia
        else
            end.setNext(no);       // adiciona nó no final da lista
        end = no;                 // atualiza referência ao nó do final
        size++;
    }

}

/**
 * Fila.java
 * Classe que implementa uma Fila
 * 18 julho de 2003
 * SCE - PGCC - UFSC
 * @author Augusto César
 */

//package container.remocao;

import container.*;

public abstract class Fila{

    private Node front, end;          // referencia para os nós do inicio e final
    private int size;                 // tamanho da pilha

    public Fila(){

```

```

        size = 0;
        front = null;
        end = null;
    }
    public int size(){
        return size;
    }
    public boolean isEmpty(){
        if (front==null)
            return true;
        return false;
    }
    public Object front(){
        if(isEmpty())
            throw new RuntimeException("Fila está vazia.");
        return front.getElement();
    }
    public Object dequeue()throws RuntimeException{
        Object elem;
        if (size==0)
            throw new RuntimeException("Fila está vazia.");
        elem = front.getElement();
        front = front.getNext();
        size--;
        if (size==0)
            end = null;    // a fila está vazia agora
        return elem;
    }
}

```