

Lista em Java (duplamente encadeada)

A lista possui duas classes: a classe **No** e a classe **Lista**. A classe **No** é para instanciar objetos que representarão uma caixinha, como se fosse uma lista de ponteiro (lista dinâmica) mostrado na Figura 1. A classe **No** possui os seguintes métodos:

- método **No getAnt()**: devolve o endereço do objeto **No** anterior a partir de algum objeto **No** determinado. Se não houver algum objeto **No** anterior o método, retornará **null**. Um objeto **No** é inicializado atribuindo **null** ao **ant** e **prox** (isto é feito no construtor da classe **No**);
- método **No getProx()**: similar ao método **getAnt**, mas para devolver o endereço do objeto **No** posterior a partir de algum objeto **No** determinado;
- método **void setAnt(No ant)**: estabelece o endereço de algum objeto **No** ao atributo **this.ant**;
- método **void setProx(No prox)**: estabelece o endereço de algum objeto **No** ao atributo **this.prox**;
- método **int getInfo()**: retorna o valor da informação do nó;
- **void setInfo(int info)**: atribui o valor da variável **info** à informação do nó.

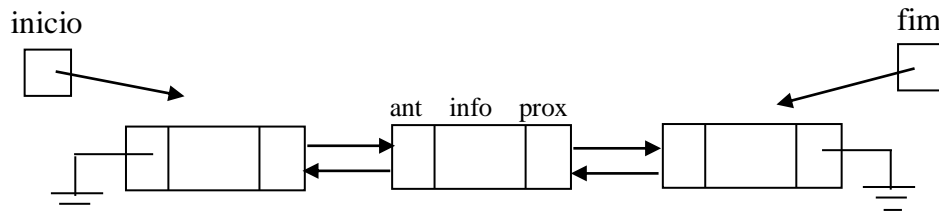


Figura 1: Representação de uma lista dinâmica duplamente encadeada.

A classe **Lista** possui os seguintes métodos:

- método **void inicializa()**;
- método **void inserirNoInicio(int info)**;
- método **void inserirNoFinal(int info)**;
- método **void exibir()**;
- método **No busca_exaustiva(int info)**;
- método **void remover(int info)**.

A Figura 2 mostra um diagrama com as classes **No** e **Lista**, juntamente com a classe **Aplicacao** que executará os métodos de um objeto instanciado a partir da classe **Lista**.

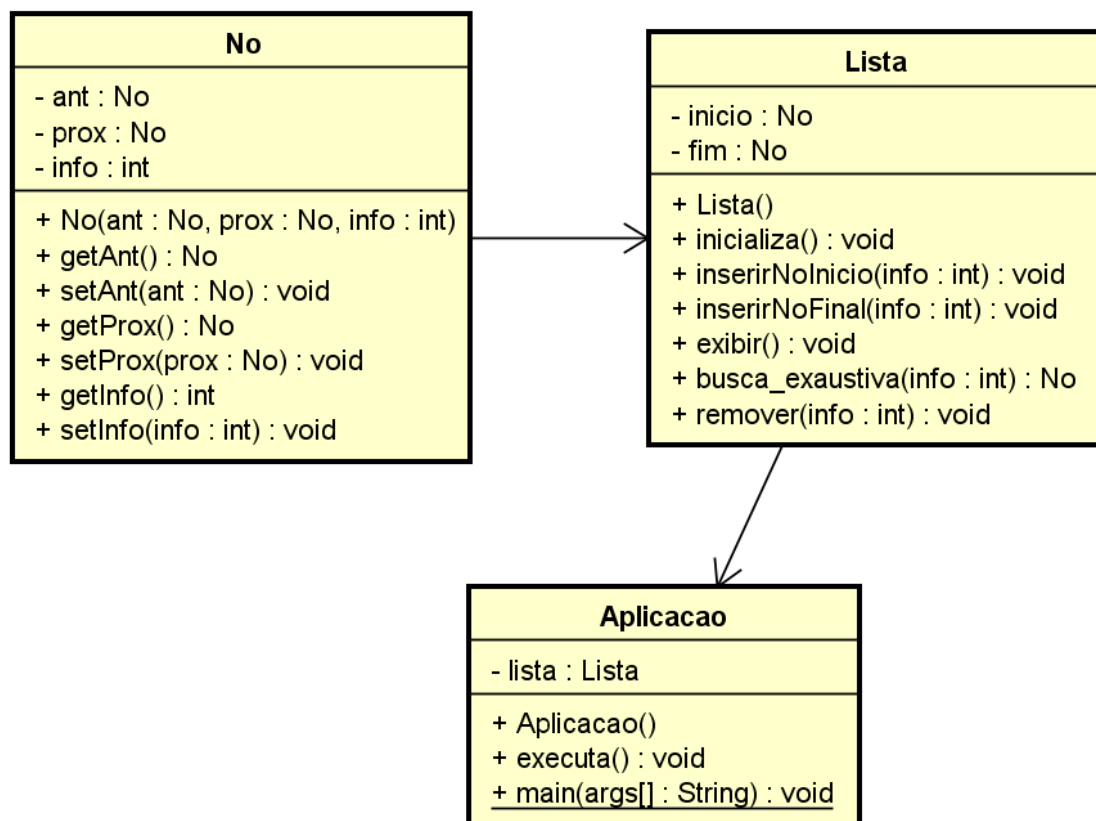


Figura 2: Diagrama de classes da Lista encadeada em Java.