



O objetivo deste exercício é implementar um TAD Lista por meio de uma lista simplesmente encadeada. Para tal, você terá de implementar os métodos das classes abaixo (use **copy** e **paste** para copiar o código).

Pacote lista

Crie um pacote chamado **lista**, seguindo os seguintes passos:

1. Selecione, com o botão direito do mouse, a pasta **src** do projeto criado para este exercício
2. Escolha **New => Package**
3. Digite o nome **lista** no campo **Name** e pressione **Finish**
4. O pacote **lista** será criado

```
class No {
    private Object elem;
    private No prox;

    public No(Object e, No p) {
        elem=e;
        prox=p;
    }
    public Object getElem() {
        //insira o código da função
    }
    public No getProx() {
        //insira o código da função
    }
    public void setProx(No o) {
        //insira o código da função
    }
}
```

```
public class Lista {
    private int tam=0;
    private No ini=null;
    private No fin=null;
    private No corr=null;

    public boolean vazio() {
        //insira o código da função
    }
    public void insIni(Object x) {
        //insira o código da função
    }
    public void insFin(Object x) {
        //insira o código da função
    }
}
```



```
public Object retIni()      {
    //insira o código da função
}
public Object retFin()      {
    //insira o código da função
}
public void posIni() {
    //insira o código da função
}
public Object prox() {
    //insira o código da função
}
}
```

Operações da Classe Lista

public boolean vazio() – retorna **true** se a lista estiver vazia e **false** caso contrário.

public void insIni(Object x) – insere o objeto **x** no início da lista.

public void insFin(Object x) – insere o objeto **x** no final da lista.

public Object retIni() – retira o primeiro elemento da lista e retorna uma referência o mesmo. Caso a lista esteja vazia o método deve retornar **null**.

public Object retFin() – retira o último elemento da lista e retorna uma referência o mesmo. Caso a lista esteja vazia o método deve retornar **null**.

public void posIni() – o primeiro elemento da lista passa a ser o elemento corrente (variável **corr**). Caso a lista esteja vazia **corr** irá conter o valor **null** após a execução desta operação.

public Object prox() – retorna o objeto armazenado no nó corrente (variável **corr**) e faz com que a variável **corr** referencie o próximo nó da lista. Caso a variável **corr** contenha o valor **null** esta operação deverá retornar o valor **null**.

Variáveis de Instância da Classe Lista

private int tam – guarda a quantidade de elementos armazenados na lista.

private No ini – guarda uma referência para o primeiro nó da lista. Se a lista estiver vazia **ini** será igual a **null**.

private No fin – guarda uma referência para o último nó da lista. Se a lista estiver vazia **fin** será igual a **null**.

private No corr – guarda uma referência para o nó corrente da lista. Se a lista estiver vazia **corr** será igual a **null**. Antes que o método **prox()** possa ser usado pela primeira vez é necessário que o nó corrente seja *inicializado* por meio do método **posIni()**.

Classe Principal (pacote default)



INF1636 – Programação Orientada a Objetos

Prof. Ivan Mathias Filho

Exercício 4.3

Para testar o TAD Lista, utilize a classe **Vetor**, elaborada no exercício anterior. Insira elementos no início e no final da lista, posicione o cursor no início da mesma (**Lista.posIni()**) e percorra a lista até o final (**Lista.prox()**), imprimindo as coordenadas dos mesmos.

Observação: tente criar um objeto da classe **No** no interior do método **main()**.