**2. Atividade de Lista Duplamente Encadeada**

Objetivo:

Construir um sistema em Java que receba, como parâmetro, o endereço do primeiro nodo de uma lista encadeada e um valor. O sistema deve retornar:

* O número total de nodos da lista.
* O número de nodos da lista que possuem o conteúdo igual ao valor passado como parâmetro e suas respectivas posições na lista.
* O número de nodos que possuem em seu conteúdo valores maiores do que o valor passado como parâmetro.

**Atividade 1: Implementar a Classe Nodo-**Descrição: Criar uma classe chamada Nodo que representará um nodo da lista encadeada.Tarefa:

* Declarar um campo int dado para armazenar o valor do nodo.
* Declarar um campo Nodo proximo para referenciar o próximo nodo na lista.
* Implementar um construtor que inicializa o campo dado e define proximo como null.

**Atividade 2: Implementar a Classe ListaEncadeada-** Descrição: Criar uma classe chamada ListaEncadeada que implementará a lista encadeada.Tarefa:

* Declarar um campo Nodo cabeca para apontar para o primeiro nodo da lista.
* Implementar os seguintes métodos:
* adicionar(int dado): Adiciona um novo nodo ao final da lista.
* imprimirLista(): Percorre e imprime todos os nodos da lista.
* analisarLista(int valor): Analisa a lista para retornar as informações solicitadas.

**Atividade 3: Implementar o Método adicionar-**Descrição: Implementar o método adicionar na classe ListaEncadeada.Tarefa:

* Verificar se a lista está vazia (i.e., cabeca é null). Se estiver, criar um **novo nodo** e definir cabeça como este **novo nodo**.
* Caso contrário, percorrer a lista até encontrar o último nodo e adicionar o novo nodo ao final.

**Atividade 4: Implementar o Método imprimirLista-**Descrição: Implementar o método imprimirLista na classe ListaEncadeada.Tarefa:

* Percorrer a lista desde a cabeca, imprimindo o valor de cada nodo até o final da lista.

**Atividade 5: Implementar o Método analisarLista**

Descrição: Implementar o método analisarLista na classe ListaEncadeada.Tarefa:

* Receber como parâmetros o endereço do primeiro nodo da lista e um valor.
* Calcular e retornar:
* O número total de nodos na lista.
* O número de nodos que possuem o valor igual ao passado como parâmetro e suas respectivas posições.
* O número de nodos que possuem valores maiores do que o valor passado como parâmetro.

**Atividade 6: Testar a Implementação-**Descrição: Escrever um método main para testar a implementação das classes Nodo e ListaEncadeada.Tarefa:

* Criar uma instância da classe ListaEncadeada.
* Adicionar alguns valores à lista.
* Analisar a lista para um valor específico.
* Imprimir os resultados da análise.