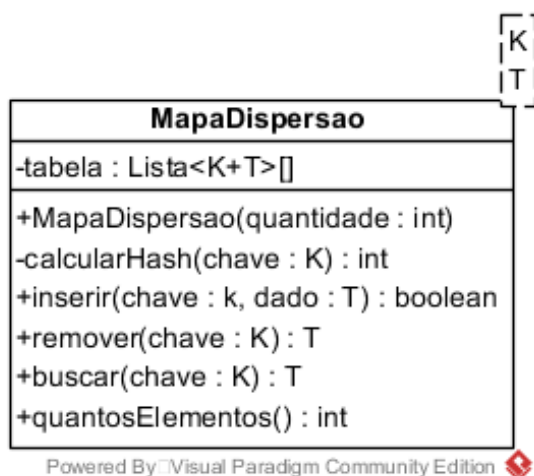


Implemente um mapa de dispersão que utilize tratamento de colisões por endereçamento separado (listas encadeadas), conforme diagrama abaixo:



- O construtor `MapaDispersao(int quantidade)` deve criar um mapa com vetor encapsulado, cujo tamanho será calculado com base no argumento `quantidade`, que é a quantidade estimada de elementos a serem inseridos. Considere as boas práticas para determinar o tamanho deste vetor.
- O método privado `calcularHash()` deve delegar para a classe `K` o cálculo do `hash`, reusando o método `hashCode()` do objeto recebido como argumento (chave). Entretanto, o método `calcularHash()` deverá compactar o valor retornado por `hashCode()` para um intervalo aceitável para ser armazenado no vetor `tabela`.
- O método `inserir(K, T)` deve armazenar o objeto `T`, fornecido como argumento, no mapa de dispersão de acordo com o valor da chave (`K`). O retorno do método é um booleano: `true` caso a inserção seja bem sucedida; `false` caso já exista a chave no mapa e por isto não seja inserido o objeto `T` no mapa.
- O método `remover(K)` deve remover do mapa de dispersão o objeto que possui a mesma chave de busca da chave fornecida como argumento, retornando este objeto. Caso não localize, deve retornar `null`.
- O método `buscar(K)` deve procurar no mapa de dispersão um objeto que possua chave de busca igual à fornecida como argumento. Como resultado do seu processamento, o método deve retornar o objeto localizado ou `null` caso não localize.
- O método `quantosElementos()` deve retornar a quantidade atual de elementos inseridos no mapa.

Reforçando que não poderá ser utilizada nenhuma classe da JCF.

Não utilize pacotes, ou seja, deixe todas as suas classes no pacote `default`.

Após implementar o mapa de dispersão, crie uma classe `Principal` contendo o método `main()` para testar e demonstrar o funcionamento da estrutura de dados implementada.

Para testar o funcionamento da inserção, busca e remoção de objetos, utilize a classe Veiculo implementada no exercício da Lista 8 e cuja chave é a placa do Veículo:

Veículo
- placa: String - modelo: String - ano: int - proprietário: String
+ toString() : String

Por exemplo, seu método `main()` poderia ter o seguinte trecho de código:

```
Veiculo v = new Veiculo();  
v.setPlaca("AAA-0123");  
v.setModelo("Hyundai Creta");  
v.setAno(2017);  
v.setProprietario("João da Silva");  
MapaDispersao<String,Veiculo> mp = new MapaDispersao(2000);  
mp.inserir(v.getPlaca(), v);
```

Trabalho em duplas ou trios para entrega até **06/06 (3ª feira) às 18h30**.

Entregue apenas as classes (arquivos .java, sem compactar) por meio da tarefa no AVA (Unidade 4).

Os nomes dos membros da equipe devem figurar como comentário em cada código-fonte e devem ser informados ao professor (marcel@furb.br) para constarem com grupo no AVA.