

6. Considere a existência de duas listas simplesmente encadeadas em que cada nó encadeado é representado pelo tipo NoLista abaixo.

```
typedef struct noLista NoLista;  
struct noLista  
{   int codigo;  
    char nome[51];  
    NoLista *prox;  
};
```

Escreva a função **testaContida** que recebe duas listas simplesmente encadeadas (ou seja, os endereços dos primeiros nós das duas listas) e retorna 1 se a primeira lista está totalmente contida na segunda lista. E retorna 0 em caso contrário. NÃO USE FUNÇÕES AUXILIARES (exceto as disponibilizadas pelas bibliotecas padrão de C).

Por exemplo, se os elementos da lista 1 fossem:

[888,"lia"], [222,"leo"], [777,"rosa"], [333, "rui"]

E os elementos da lista 2 fossem:

[111, "edu"], [222,"leo"], [333, "rui"], [446, "cris"], [558, "lara"], [777,"rosa"], [888,"lia"], [999,"bia"]

A função retorna 1.

Por exemplo, se os elementos da lista 1 fossem:

[888,"lia"], [222,"leo"], [554, "vera"], [777,"rosa"], [333, "rui"]

E os elementos da lista 2 fossem:

[111, "edu"], [222,"leo"], [333, "rui"], [446, "cris"], [666,"cadu"], [777,"rosa"], [888,"lia"], [999,"bia"]

A função retorna 0.

Considere que nas listas recebidas, nem todos os valores da lista 2 estão na lista 1 e vice-versa. Considere também que não há repetições de "codigo" numa mesma lista e que um determinado "codigo" corresponderá sempre a um mesmo nome. Observe que as listas recebidas não são ordenadas.