

Considere a existência de uma lista simplesmente encadeada com os dados dos livros de uma editora. A lista encontra-se ordenada crescentemente pelos anos dos livros.

O tipo estruturado que representa um nó dessa lista encadeada é o tipo NoDaLista apresentado a seguir:

```
typedef struct noDaLista NoDaLista;
struct noDaLista
{
    char titulo[51];
    char autor[51];
    int ano;
    int quantidade;
    NoDaLista *prox;
};
```

Escreva a função `separaUmAno`, que recebe a lista (ou seja, o endereço do primeiro nó da lista) e um ano, e MODIFICANDO a lista original, retorna uma (sub)lista encadeada somente com os livros daquele ano. Não devem ser criados novos nós. O trecho correspondente aos livros desse ano deve ser “cortado” da lista original. A lista original deve ser “emendada”, e o endereço do primeiro nó do trecho (sublista) arrancado deve ser retornado (sendo necessário colocar NULL no campo `prox` do último nó dessa (sub)lista). A sua função deve levar em conta a ordenação da lista original. A função retorna o endereço do primeiro nó da (sub) lista resultante.

Para facilitar, considere que o ano solicitado nunca é o do primeiro livro ou do último livro. Se não existir nenhum livro para o ano recebido, a função retorna NULL.