

Lista 3 - final

Exercício 1. Uma lista circular, é uma lista ligada onde o ultimo no aponta para o primeiro, em vez de apontar para NULL.

Dado:

```
struct regis {
    int dado;
    struct regis *prox;
}
typedef struct regis * APO
```

escreva função `tira_menor` que dado um apontador para uma lista circular, não vazia, remove da lista o menor numero armazenado.

Exercício 2. Dado:

```
struct regis {
    int dado;
    struct regis *prox;
}
typedef struct regis * APO
```

escreva a função **recursiva crescente** que dado um apontador para uma lista retorna 1 se a lista estiver em ordem crescente, e 0 se não . (Se a função não for recursiva, a questão valerá no máximo 50%).

Exercício 3. Suponha o seguinte definido

```
struct regis {
    int dado;
    struct regis *prox;
}
typedef struct regis * APO
```

Escreva uma função que remove o “ultimo” elemento de uma lista circular (o “ultimo” elemento da lista aponta para o “primeiro” em vez de apontar para NULL como no caso das listas tradicionais).

A declaração da função é

```
void removeult (APO *inicio);
```

onde `inicio` é um apontador para o primeiro elemento da lista circular.