

```

/*
 * Faça uma biblioteca para as definições
 * abaixo especificadas.
 */

typedef int TipoChave;
typedef int TipoValor;
struct TipoListaCircular
{
    TipoChave chave;
    TipoValor valorQualquer;
    struct TipoListaCircular *prox;
};

typedef struct TipoListaCircular TipoListaCircular;

/*=====>PROCEDIMENTOS BÁSICOS DE LISTAS CIRCULAR
/* -----> Insercao
 * Insere novo nó no início de uma lista encadeada circular.
 * A referência de ponteiro para o primeiro nó e os
 * valores dos campos do nó são dados.
 * Devolve endereço do nó recém inserido
 * ou NULL em caso de insucesso.
 */

TipoListaCircular *insereInicioListaCircular(TipoListaCircular **prim,
TipoChave chave, TipoValor valor);

/* -----> Remove nó por valor de chave
 * Remove nó cujo valor chave seja igual a 'chave'
 * Mantêm lista inalterada caso este não exista.
 */

void removeNo(TipoListaCircular **prim, TipoChave chave);

/* -----> Cria cópia
 * Cria uma nova lista cujos nós tem valor chave par
 * Devolve o ponteiro para
 * o primeiro nó da nova lista.
 */

TipoListaCircular *copiaListaPar(TipoListaCircular *prim);

```