

```
// IED-001 (Prof. Dr. Silvio do Lago Pereira)
```

```
// -----  
// Exemplo 1  
// -----
```

```
typedef int Item;  
typedef struct no {  
    Item item;  
    struct no *prox;  
} *Lista;
```

```
// -----  
// Exemplo 2  
// -----
```

```
Lista no(Item x, Lista p) {  
    Lista n = malloc(sizeof(struct no));  
    n->item = x;  
    n->prox = p;  
    return n;  
}
```

```
// -----  
// Exemplo 4  
// -----
```

```
void ins(Item x, Lista *L) {  
    while( *L != NULL && (*L)->item < x )  
        L = &(*L)->prox;  
    *L = no(x,*L);  
}
```

```
// -----  
// Exercício 1  
// -----
```

```
#include <stdio.h>  
#include <stdlib.h>  
...  
int main(void) {  
    Lista I = NULL;  
    ins(4,&I);  
    ins(1,&I);  
    ins(3,&I);  
    ins(5,&I);  
    ins(2,&I);  
    exhibe(I);  
    return 0;  
}
```

```
// -----  
// Exemplo 5  
// -----
```

```
void rem(Item x, Lista *L) {  
    while( *L != NULL && (*L)->item < x )  
        L = &(*L)->prox;  
    if( *L == NULL || (*L)->item > x ) return;  
    Lista n = *L;  
    *L = n->prox;  
    free(n);  
}
```

```
// -----  
// Exercício 7  
// -----  
  
#include <stdio.h>  
#include <stdlib.h>  
  
...  
  
int main(void) {  
    Lista I = NULL;  
    ins(4,&I); ins(1,&I); ins(3,&I); ins(5,&I); ins(2,&I);  
    rem(3,&I);  
    exhibe(I);  
    return 0;  
}  
  
// -----  
// Exemplo 6  
// -----  
  
int pert(Item x, Lista L) {  
    while( L != NULL && L->item < x )  
        L = L->prox;  
    return (L != NULL && L->item == x);  
}  
  
// -----  
// Exercício 9  
// -----  
  
#include <stdio.h>  
#include <stdlib.h>  
  
...  
  
int main(void) {  
    Lista I = NULL;  
    ins(4,&I);  
    ins(1,&I);  
    ins(3,&I);  
    ins(2,&I);  
    printf("%d\n", pert(5,I));  
    printf("%d\n", pert(3,I));  
    return 0;  
}
```