

## Exemplo: C

```

typedef
struct elem{
    int chave, valor;
    struct elem * next;
}element_pt;

element_pt * lista;

void inserir(elemento_pt *lista, int chave, int valor)
{
    elemento_pt *p;
    p = (elemento_pt*) malloc(sizeof
        (elemento_pt));
    p->chave = chave;  p->valor = valor;
    p->next = lista;  lista = p;
};

```

## Exemplo: C

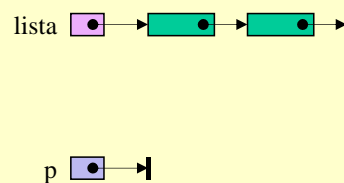
```

typedef
struct elem{
    int chave, valor;
    struct elem * next;
}element_pt;

element_pt * lista;

void inserir(elemento_pt *lista, int chave, int valor)
{
    elemento_pt *p;
    p = (elemento_pt*) malloc(sizeof
        (elemento_pt));
    p->chave = chave;  p->valor = valor;
    p->next = lista;  lista = p;
};

```

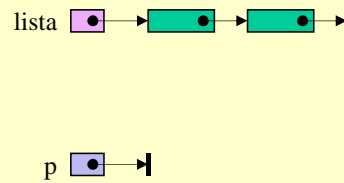


## Exemplo: C

```
typedef
struct elem{
    int chave, valor;
    struct elem * next;
}element_pt;
```

```
element_pt * lista;
```

```
void inserir(elemento_pt *lista, int chave, int valor)
{
    elemento_pt *p;
    p = (elemento_pt*) malloc(sizeof
        (elemento_pt));
    p->chave = chave; p->valor = valor;
    p->next = lista; lista = p;
};
```

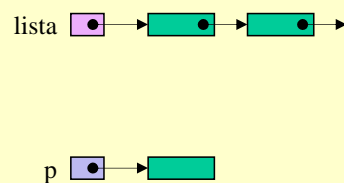


## Exemplo: C

```
typedef
struct elem{
    int chave, valor;
    struct elem * next;
}element_pt;
```

```
element_pt * lista;
```

```
void inserir(elemento_pt *lista, int chave, int valor)
{
    elemento_pt *p;
    p = (elemento_pt*) malloc(sizeof
        (elemento_pt));
    p->chave = chave; p->valor = valor;
    p->next = lista; lista = p;
};
```

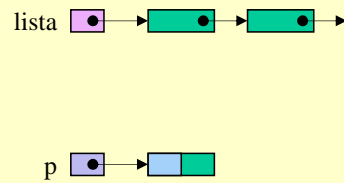


## Exemplo: C

```
typedef
struct elem{
    int chave, valor;
    struct elem * next;
}element_pt;
```

```
element_pt * lista;
```

```
void inserir(elemento_pt *lista, int chave, int valor)
{
    elemento_pt *p;
    p = (elemento_pt*) malloc(sizeof
        (elemento_pt));
    p->chave = chave; p->valor = valor;
    p->next = lista; lista = p;
};
```

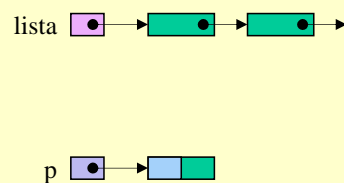


## Exemplo: C

```
typedef
struct elem{
    int chave, valor;
    struct elem * next;
}element_pt;
```

```
element_pt * lista;
```

```
void inserir(elemento_pt *lista, int chave, int valor)
{
    elemento_pt *p;
    p = (elemento_pt*) malloc(sizeof
        (elemento_pt));
    p->chave = chave; p->valor = valor;
    p->next = lista; lista = p;
};
```

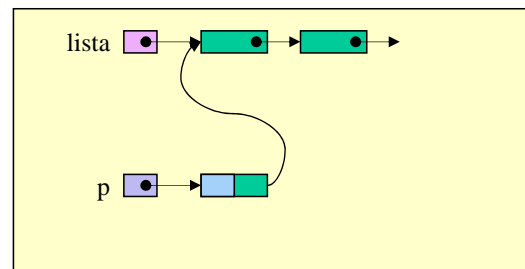


## Exemplo: C

```
typedef
struct elem{
    int chave, valor;
    struct elem * next;
}element_pt;
```

```
element_pt * lista;
```

```
void inserir(elemento_pt *lista, int chave, int valor)
{
    elemento_pt *p;
    p = (elemento_pt*) malloc(sizeof
        (elemento_pt));
    p->chave = chave;  p->valor = valor;
    p->next = lista;  lista = p;
};
```

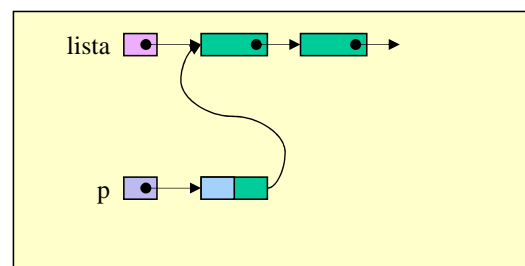


## Exemplo: C

```
typedef
struct elem{
    int chave, valor;
    struct elem * next;
}element_pt;
```

```
element_pt * lista;
```

```
void inserir(elemento_pt *lista, int chave, int valor)
{
    elemento_pt *p;
    p = (elemento_pt*) malloc(sizeof
        (elemento_pt));
    p->chave = chave;  p->valor = valor;
    p->next = lista;  lista = p;
};
```

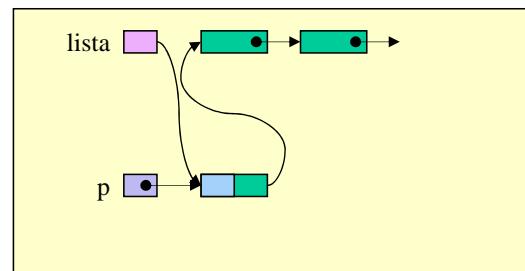


## Exemplo: C

```
typedef
struct elem{
    int chave, valor;
    struct elem * next;
}element_pt;
```

```
element_pt * lista;
```

```
void inserir(elemento_pt *lista, int chave, int valor)
{
    elemento_pt *p;
    p = (elemento_pt*) malloc(sizeof
        (elemento_pt));
    p->chave = chave; p->valor = valor;
    p->next = lista; lista = p;
};
```

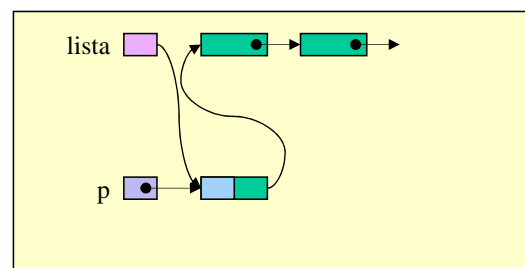


## Exemplo: C

```
typedef
struct elem{
    int chave, valor;
    struct elem * next;
}element_pt;
```

```
element_pt * lista;
```

```
void inserir(elemento_pt *lista, int chave, int valor)
{
    elemento_pt *p;
    p = (elemento_pt*) malloc(sizeof
        (elemento_pt));
    p->chave = chave; p->valor = valor;
    p->next = lista; lista = p;
};
```

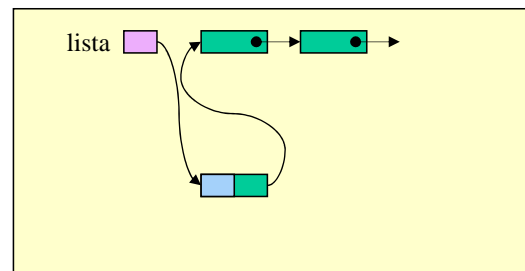


**Exemplo: C**

```
typedef
struct elem{
    int chave, valor;
    struct elem * next;
}element_pt;
```

```
element_pt * lista;
```

```
void inserir(elemento_pt *lista, int chave, int valor)
{
    elemento_pt *p;
    p = (elemento_pt*) malloc(sizeof
        (elemento_pt));
    p->chave = chave;  p->valor = valor;
    p->next = lista;  lista = p;
};
```

**Exemplo: C**

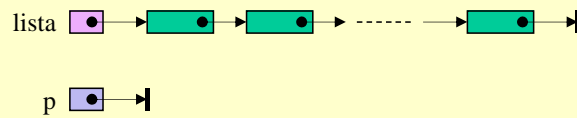
```
void remover(elemento_pt *lista)
{
    elemento_pt *p;
    while (lista != NULL)
    {
        p = lista;
        lista = lista->next;
        free(p);
    };
};
```

## Exemplo: C

```

void remover(elemento_pt *lista)
{
    elemento_pt *p;
    while (lista != NULL)
    {
        p = lista;
        lista = lista->next;
        free(p);
    };
};

```

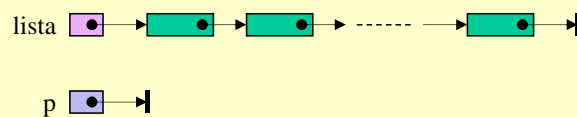


## Exemplo: C

```

void remover(elemento_pt *lista)
{
    elemento_pt *p;
    while (lista != NULL)
    {
        p = lista;
        lista = lista->next;
        free(p);
    };
};

```

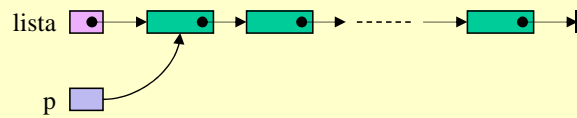


## Exemplo: C

```

void remover(elemento_pt *lista)
{
    elemento_pt *p;
    while (lista != NULL)
    {
        p = lista;
        lista = lista->next;
        free(p);
    }
};

```



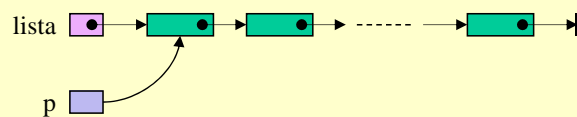
15

## Exemplo: C

```

void remover(elemento_pt *lista)
{
    elemento_pt *p;
    while (lista != NULL)
    {
        p = lista;
        lista = lista->next;
        free(p);
    }
};

```



16

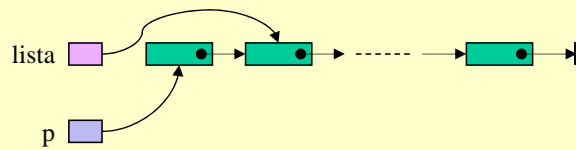


**Exemplo: C**

```

void remover(elemento_pt *lista)
{
    elemento_pt *p;
    while (lista != NULL)
    {
        p = lista;
        lista = lista->next;
        free(p);
    }
};

```



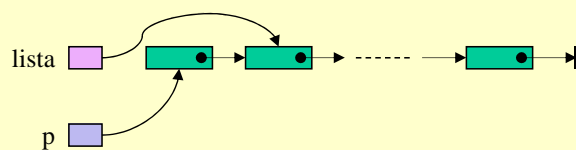
17

**Exemplo: C**

```

void remover(elemento_pt *lista)
{
    elemento_pt *p;
    while (lista != NULL)
    {
        p = lista;
        lista = lista->next;
        free(p);
    }
};

```



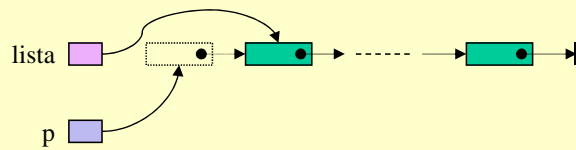
18

**Exemplo: C**

```

void remover(elemento_pt *lista)
{
    elemento_pt *p;
    while (lista != NULL)
    {
        p = lista;
        lista = lista->next;
        free(p);
    };
};

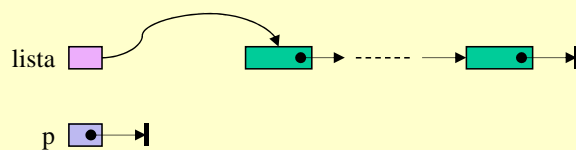
```

**Exemplo: C**

```

void remover(elemento_pt *lista)
{
    elemento_pt *p;
    while (lista != NULL)
    {
        p = lista;
        lista = lista->next;
        free(p);
    };
};

```



**Exemplo: C**

```
void remover(elemento_pt *lista)
{
    elemento_pt *p;
    while (lista != NULL)
    {
        p = lista;
        lista = lista->next;
        free(p);
    };
};
```

lista 

**Exemplo: C**

```
void esvaziar_lista(elemento_pt *lista)
{
    lista = NULL;
};
```

lista 

### Exemplo: C

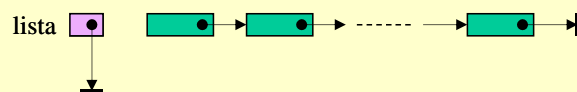
```
void esvaziar_lista(elemento_pt *lista)
{
    lista = NULL;
};
```



23

### Exemplo: C

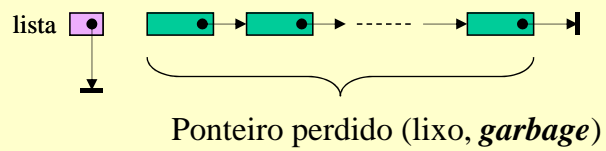
```
void esvaziar_lista(elemento_pt *lista)
{
    lista = NULL;
};
```



24

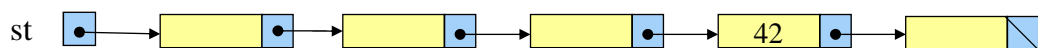
**Exemplo: C**

```
void esvaziar_lista(elemento_pt *lista)
{
    lista = NULL;
};
```



25

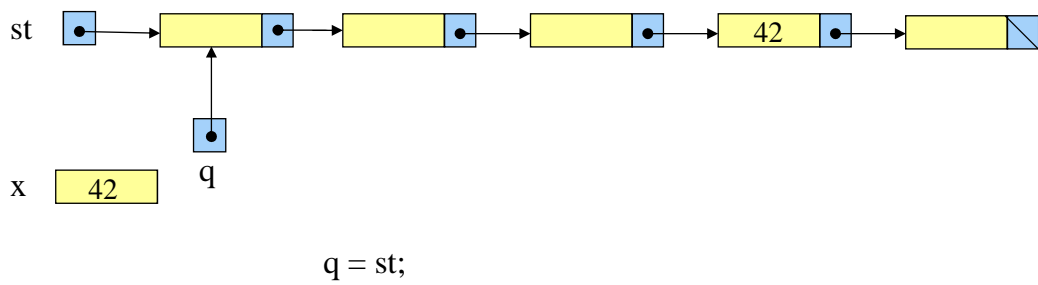
`busca(st, x)`



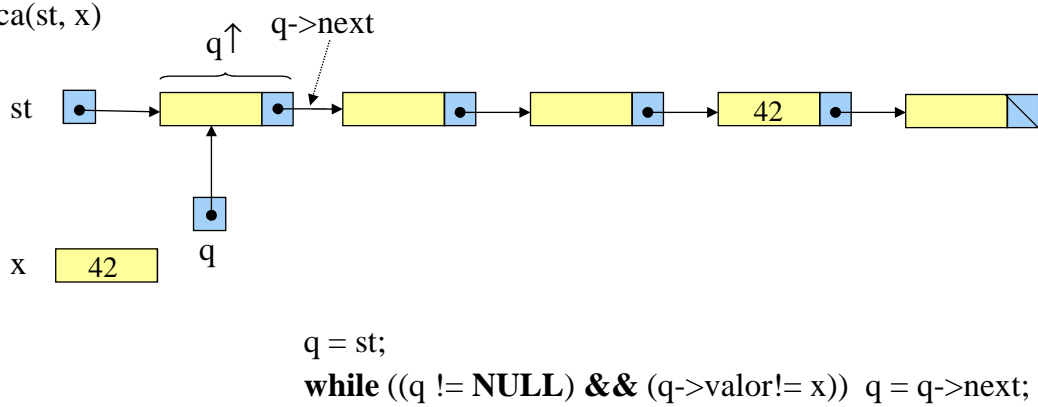
`x` 42

26

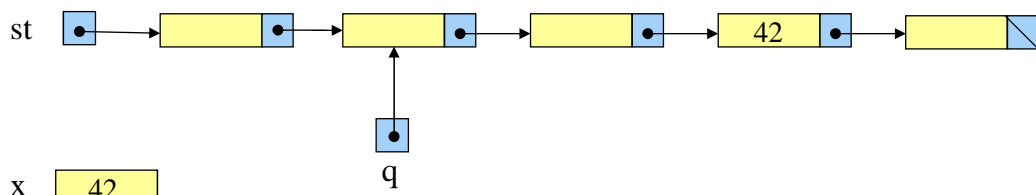
busca(st, x)



busca(st, x)



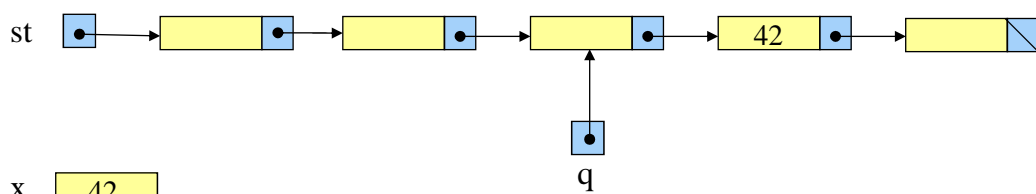
busca(st, x)



`q = st;`

**while** `((q != NULL) && (q->valor != x)) q = q->next;`

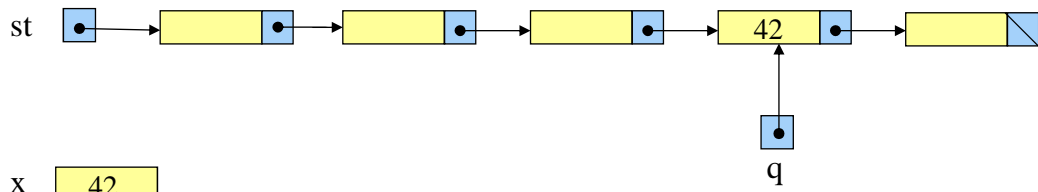
busca(st, x)



`q = st;`

**while** `((q != NULL) && (q->valor != x)) q = q->next;`

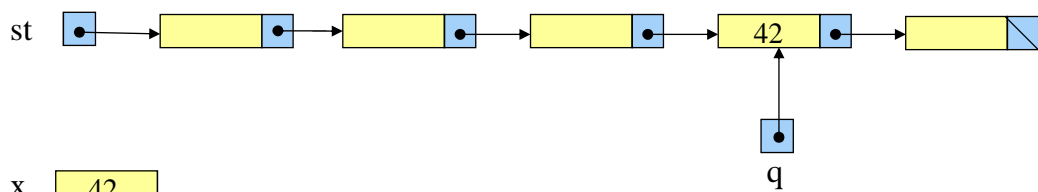
busca(st, x)



`q = st;`

**while** `((q != NULL) && (q->valor != x)) q = q->next;`

busca(st, x)



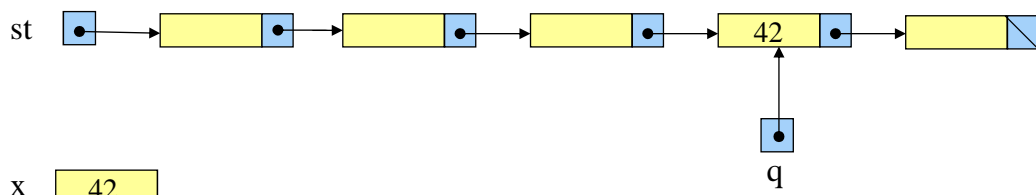
`q = st;`

**while** `((q != NULL) && (q->valor != x)) q = q->next;`

`return q;`



busca(st, x)

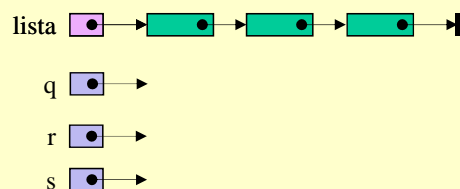


```
q = st;
while ((q != NULL) && (q->valor != x)) q = q->next;
return q;
```

busca: lista linear através de um ponteiro dinâmico (  $O(n)$  )

Exemplo: C

```
void InverteLista(elemento_pt *lista)
{
    elemento_pt *q, *r, *s ;
    q = NULL;
    r = lista;
    while (r != NULL) {
        s = r->prox;  r->prox = q;
        q = r;  r = s;
    };
    lista = q;
};
```

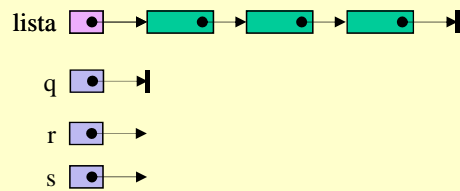


## Exemplo: C

```

void InverteLista(elemento_pt *lista)
{
    elemento_pt *q, *r, *s ;
    q = NULL;
    r = lista;
    while (r != NULL) {
        s = r->prox;  r->prox = q;
        q = r;  r = s;
    };
    lista = q;
};

```



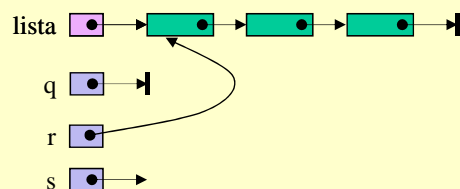
35

## Exemplo: C

```

void InverteLista(elemento_pt *lista)
{
    elemento_pt *q, *r, *s ;
    q = NULL;
    r = lista;
    while (r != NULL) {
        s = r->prox;  r->prox = q;
        q = r;  r = s;
    };
    lista = q;
};

```



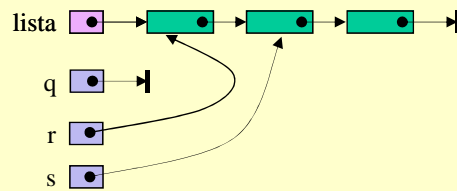
36

## Exemplo: C

```

void InverteLista(elemento_pt *lista)
{
    elemento_pt *q, *r, *s ;
    q = NULL;
    r = lista;
    while (r != NULL) {
        s = r->prox;  r->prox = q;
        q = r;  r = s;
    };
    lista = q;
};

```



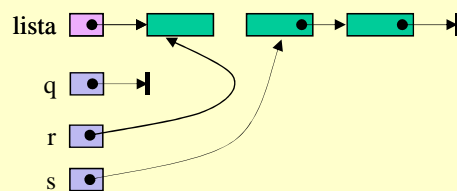
37

## Exemplo: C

```

void InverteLista(elemento_pt *lista)
{
    elemento_pt *q, *r, *s ;
    q = NULL;
    r = lista;
    while (r != NULL) {
        s = r->prox;  r->prox = q;
        q = r;  r = s;
    };
    lista = q;
};

```



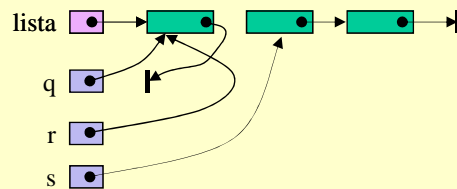
38

## Exemplo: C

```

void InverteLista(elemento_pt *lista)
{
    elemento_pt *q, *r, *s ;
    q = NULL;
    r = lista;
    while (r != NULL) {
        s = r->prox;  r->prox = q;
        q = r;  r = s;
    };
    lista = q;
};

```

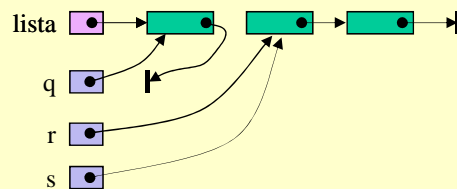


## Exemplo: C

```

void InverteLista(elemento_pt *lista)
{
    elemento_pt *q, *r, *s ;
    q = NULL;
    r = lista;
    while (r != NULL) {
        s = r->prox;  r->prox = q;
        q = r;  r = s;
    };
    lista = q;
};

```

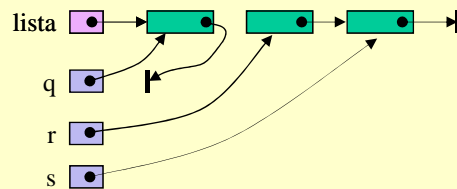


## Exemplo: C

```

void InverteLista(elemento_pt *lista)
{
    elemento_pt *q, *r, *s ;
    q = NULL;
    r = lista;
    while (r != NULL) {
        s = r->prox;  r->prox = q;
        q = r;  r = s;
    };
    lista = q;
};

```

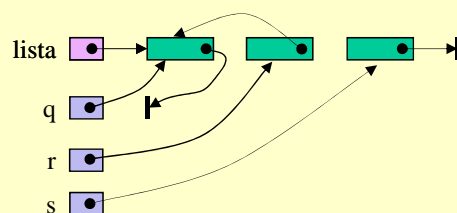


## Exemplo: C

```

void InverteLista(elemento_pt *lista)
{
    elemento_pt *q, *r, *s ;
    q = NULL;
    r = lista;
    while (r != NULL) {
        s = r->prox;  r->prox = q;
        q = r;  r = s;
    };
    lista = q;
};

```

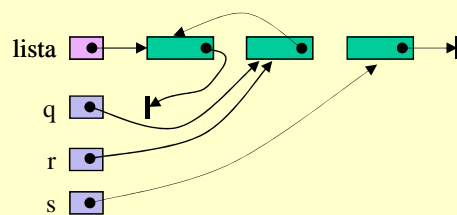


**Exemplo: C**

```

void InverteLista(elemento_pt *lista)
{
    elemento_pt *q, *r, *s ;
    q = NULL;
    r = lista;
    while (r != NULL) {
        s = r->prox;  r->prox = q;
        q = r;  r = s;
    };
    lista = q;
};

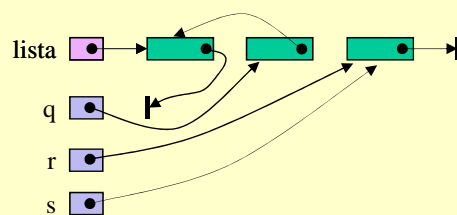
```

**Exemplo: C**

```

void InverteLista(elemento_pt *lista)
{
    elemento_pt *q, *r, *s ;
    q = NULL;
    r = lista;
    while (r != NULL) {
        s = r->prox;  r->prox = q;
        q = r;  r = s;
    };
    lista = q;
};

```

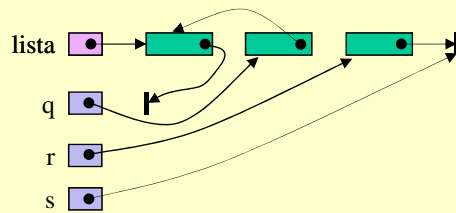


## Exemplo: C

```

void InverteLista(elemento_pt *lista)
{
    elemento_pt *q, *r, *s ;
    q = NULL;
    r = lista;
    while (r != NULL) {
        s = r->prox;  r->prox = q;
        q = r;  r = s;
    };
    lista = q;
};

```

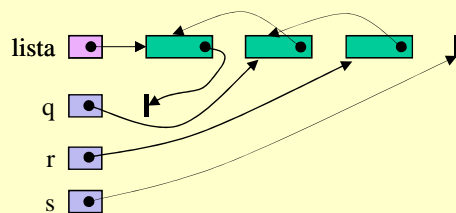


## Exemplo: C

```

void InverteLista(elemento_pt *lista)
{
    elemento_pt *q, *r, *s ;
    q = NULL;
    r = lista;
    while (r != NULL) {
        s = r->prox;  r->prox = q;
        q = r;  r = s;
    };
    lista = q;
};

```

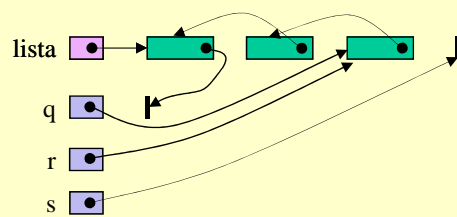


## Exemplo: C

```

void InverteLista(elemento_pt *lista)
{
    elemento_pt *q, *r, *s ;
    q = NULL;
    r = lista;
    while (r != NULL) {
        s = r->prox;  r->prox = q;
        q = r;  r = s;
    };
    lista = q;
};

```

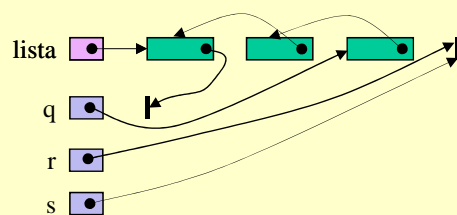


## Exemplo: C

```

void InverteLista(elemento_pt *lista)
{
    elemento_pt *q, *r, *s ;
    q = NULL;
    r = lista;
    while (r != NULL) {
        s = r->prox;  r->prox = q;
        q = r;  r = s;
    };
    lista = q;
};

```



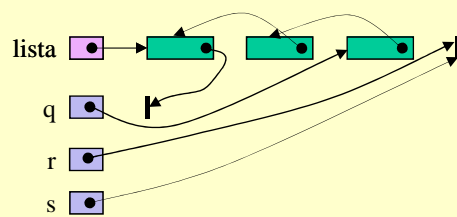


## Exemplo: C

```

void InverteLista(elemento_pt *lista)
{
    elemento_pt *q, *r, *s ;
    q = NULL;
    r = lista;
    while (r != NULL) {
        s = r->prox;  r->prox = q;
        q = r;  r = s;
    };
    lista = q;
};

```

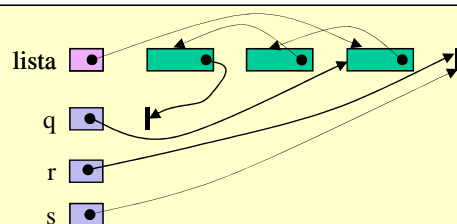


## Exemplo: C

```

void InverteLista(elemento_pt *lista)
{
    elemento_pt *q, *r, *s ;
    q = NULL;
    r = lista;
    while (r != NULL) {
        s = r->prox;  r->prox = q;
        q = r;  r = s;
    };
    lista = q;
};

```



**Exemplo: C**

```
void InverteLista(elemento_pt *lista)
{
    elemento_pt *q, *r, *s ;
    q = NULL;
    r = lista;
    while (r != NULL) {
        s = r->prox;  r->prox = q;
        q = r;  r = s;
    };
    lista = q;
};
```

