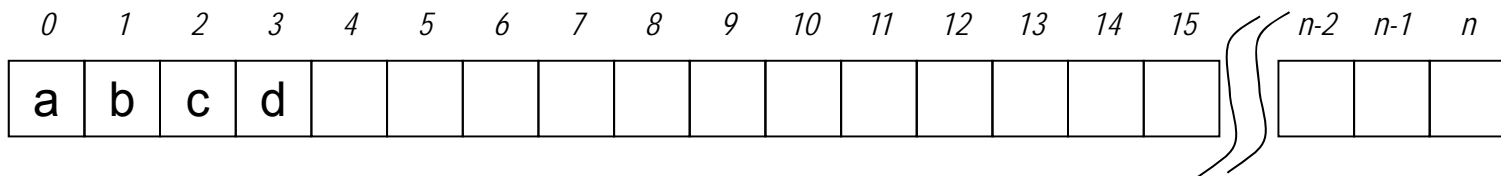
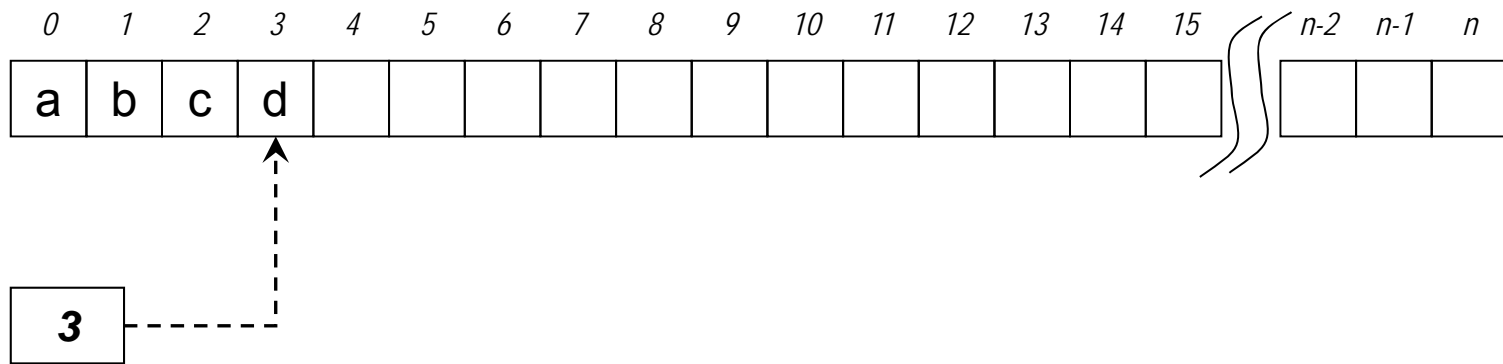


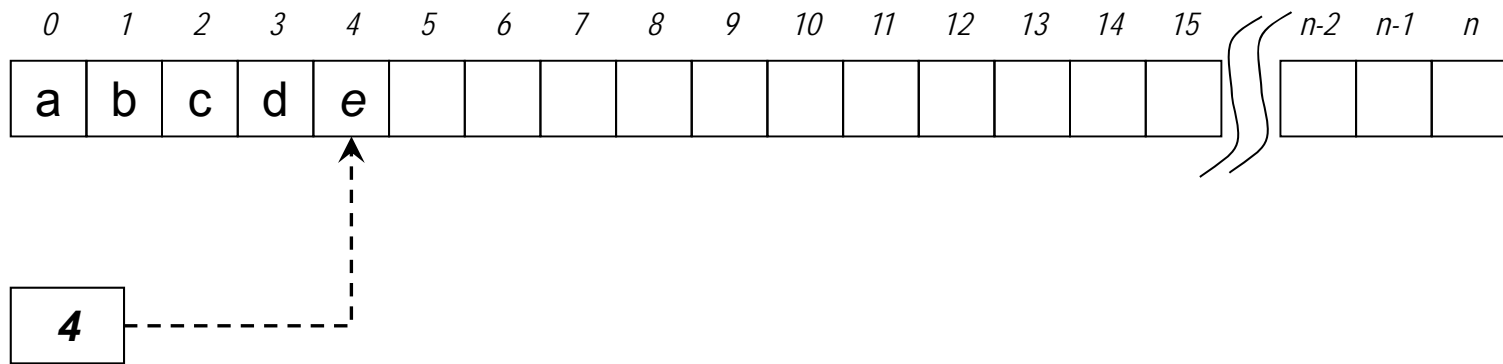
Lista



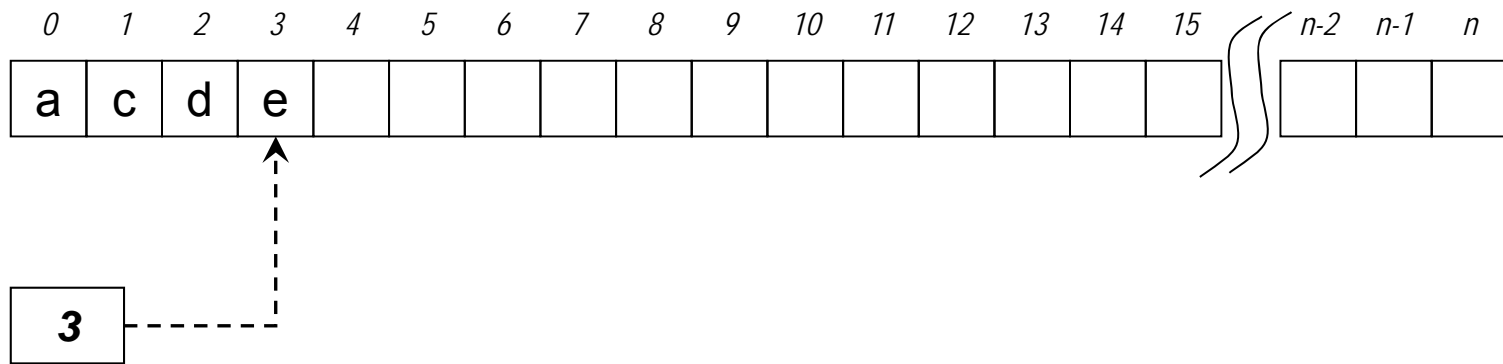
Lista



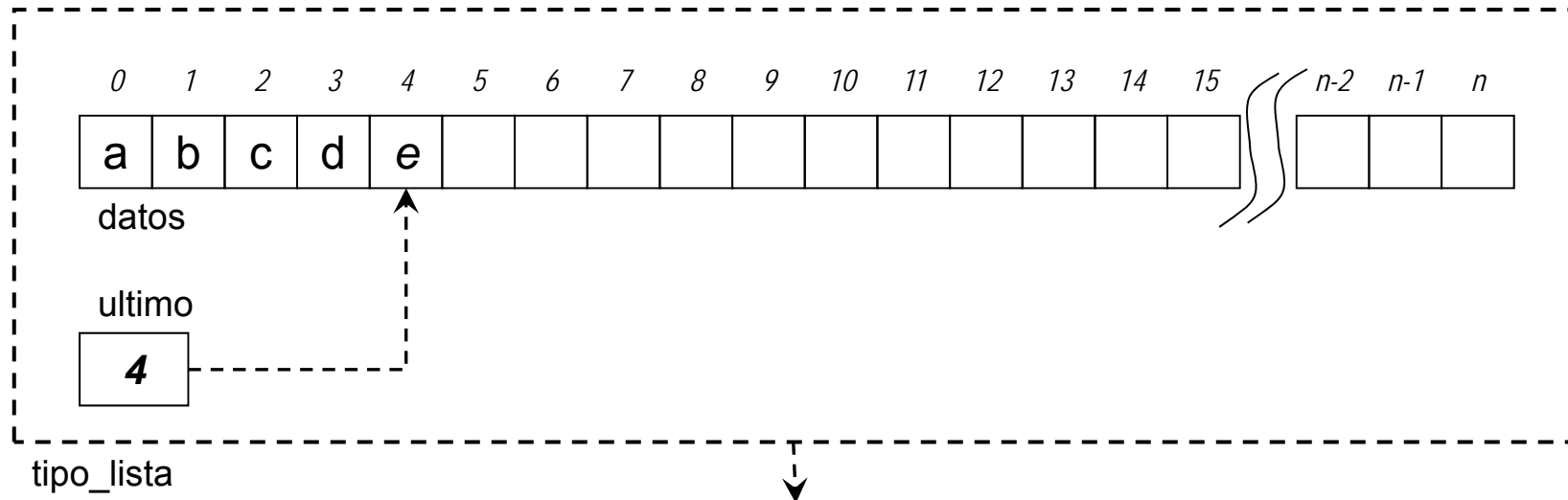
Lista



Lista



Lista



```
#define TAMMAX 500
typedef _____ tipo_dato;

typedef struct {
    tipo_dato datos[TAMMAX];
    int ultimo;
} tipo_lista;

typedef int tipo_pos;
```

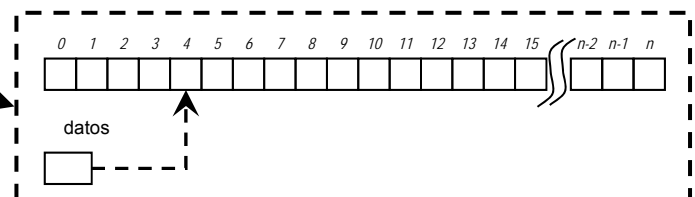
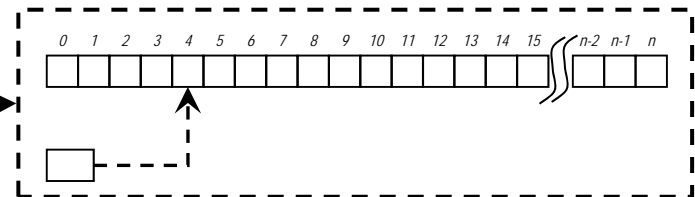
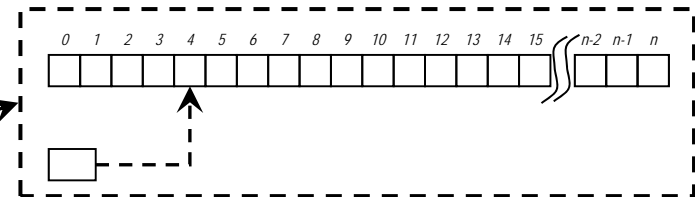
```
typedef tipo_alumno tipo_dato;
```

```
typedef struct {
    tipo_dato datos[TAMMAX];
    int ultimo;
} tipo_lista;
```

```
typedef int tipo_pos;
```

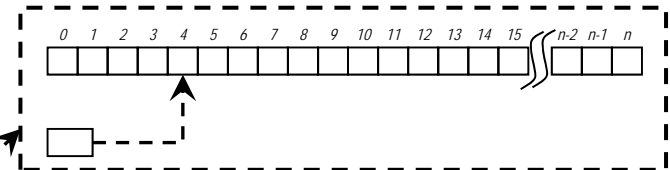
```
int main () {
    tipo_lista altas;
    tipo_lista bajas;
    tipo_lista examen;
    .
    .
    .
    altas.ultimo = -1;

    scanf ("%03s", grupo.datos[25].nombre.apm);
}
```

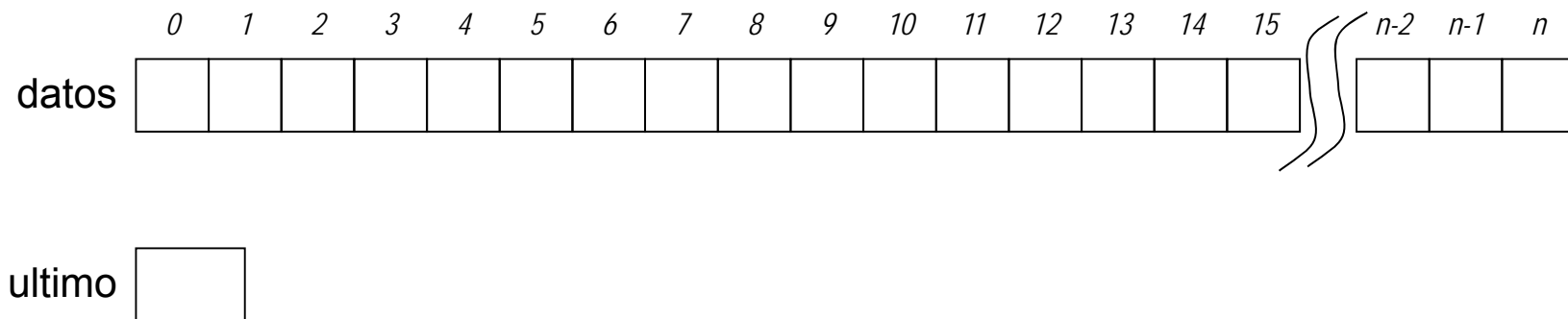


Tipo de dato:

- Conjunto de valores o elementos
- Conjunto de operaciones



Operación		Regresa
Inicializa	-	nada
Vacía	-	booleano
Llena *	-	booleano
Inserta	-	nada
Elimina	-	nada
Primero	-	posición
Ultimo	-	posición
Anterior	-	posición
Siguiente	-	posición
Localiza	-	posición
Recupera	-	elemento
Ordena	-	nada
Imprime	-	nada
Anula	-	nada



función: **inicializa**
recibe: *lista(datos[], último)*
regresa: nada

último = -1

función: **inicializa**

recibe: *lista(datos[], último)*

regresa: nada

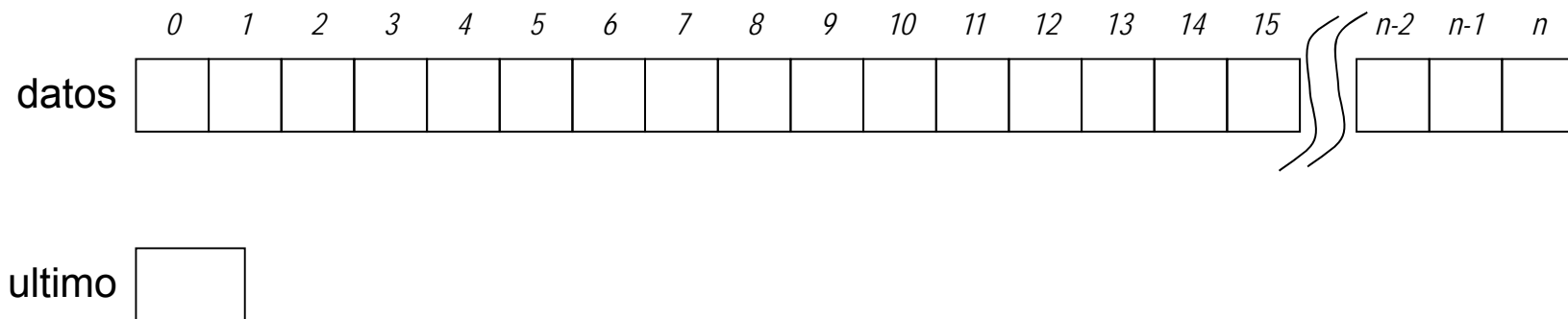
último = -1

```
typedef _____ tipo_dato;
```

```
typedef struct {  
    tipo_dato datos[TAMMAX];  
    int ultimo;  
} tipo_lista ;
```

```
typedef int tipo_pos;
```

```
void inicializa (tipo_lista *L) {  
    L->ultimo = -1;  
}
```



función: **vacía**

recibe: *lista(datos[], último)*

regresa: booleano

¿último = -1?

Sí: regresar: *verdadero*

No: regresar: *falso*

función: **vacía**

recibe: *lista(datos[], último)*

regresa: booleano

¿*último* = -1?

Sí: regresar: *verdadero*

No: regresar: *falso*

```
#define FALSE 0
```

```
#define TRUE 1
```

```
.
```

```
.
```

```
.
```

```
int vacia (tipo_lista *L) {
```

```
    if (L->ultimo == -1)
```

```
        return TRUE;
```

```
    else
```

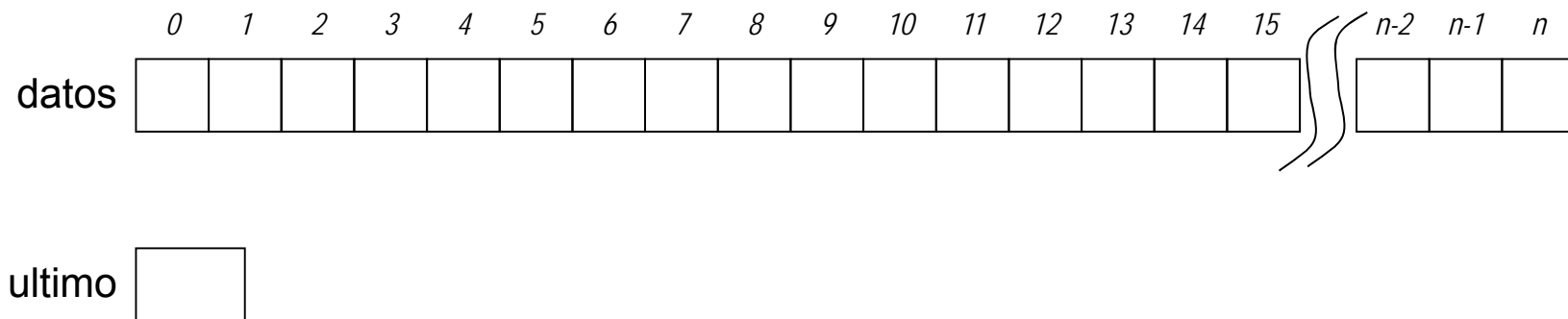
```
        return FALSE;
```

```
}
```

```
int vacia (tipo_lista *L) {
```

```
    return L->ultimo == -1;
```

```
}
```



función: **llena**

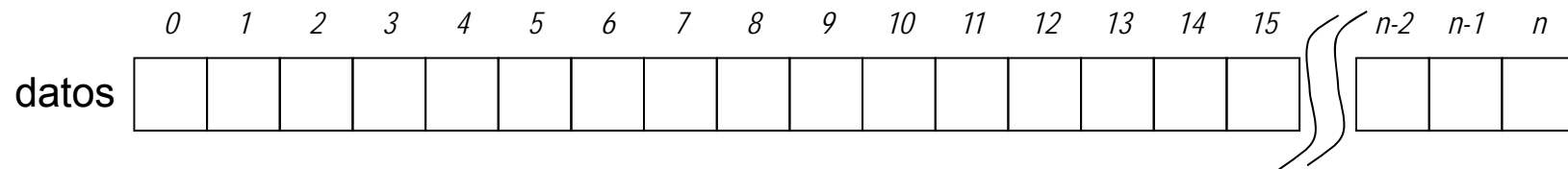
recibe: *lista(datos[], último)*

regresa: booleano

¿último = tamaño_del_arreglo - 1?

Sí: regresar: *verdadero*

No: regresar: *falso*



ultimo

función: **inserta**

recibe: *elem*, *pos*, *lista(datos[], último)*

regresa: nada

¿**llena**(*lista*) ó $pos < 0$ ó $pos > último + 1$?

Sí: desplegar mensaje de error

terminar

$i = último + 1$

mientras $i > pos$

$datos[i] = datos[i - 1]$

$i = i - 1$

fin mientras

$datos[p] = elem$

$último = último + 1$

función: **inserta**

recibe: *elem, pos, lista(datos[], último)*

regresa: nada

¿**llena**(*lista*) ó $\text{pos} < 0$ ó $\text{pos} > \text{último} + 1$?

Sí: desplegar mensaje de error
terminar

$i = \text{último} + 1$

mientras $i > \text{pos}$

$\text{datos}[i] = \text{datos}[i - 1]$

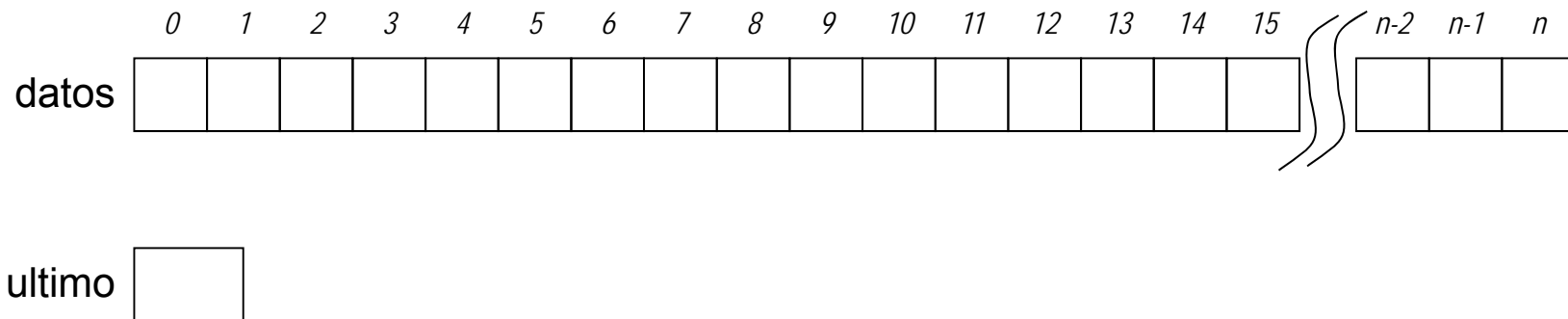
$i = i - 1$

fin mientras

$\text{datos}[p] = \text{elem}$

$\text{último} = \text{último} + 1$

```
void inserta (tipo_dato e, tipo_pos pos, tipo_lista *L) {  
    int i;  
    if (llena (L) || pos < 0 || pos > L->ultimo + 1)  
        return;  
    for (i = L->ultimo + 1; i > pos; i--)  
        L->datos[i] = L->datos[i - 1];  
    L->datos[pos] = e;  
    L->ultimo++;  
}
```



función: **elimina**

recibe: *pos*, *lista(datos[], último)*

regresa: nada

¿**vacía**(*lista*) ó $pos < 0$ ó $pos > último$?

Sí: desplegar mensaje de error
terminar

$i = pos$

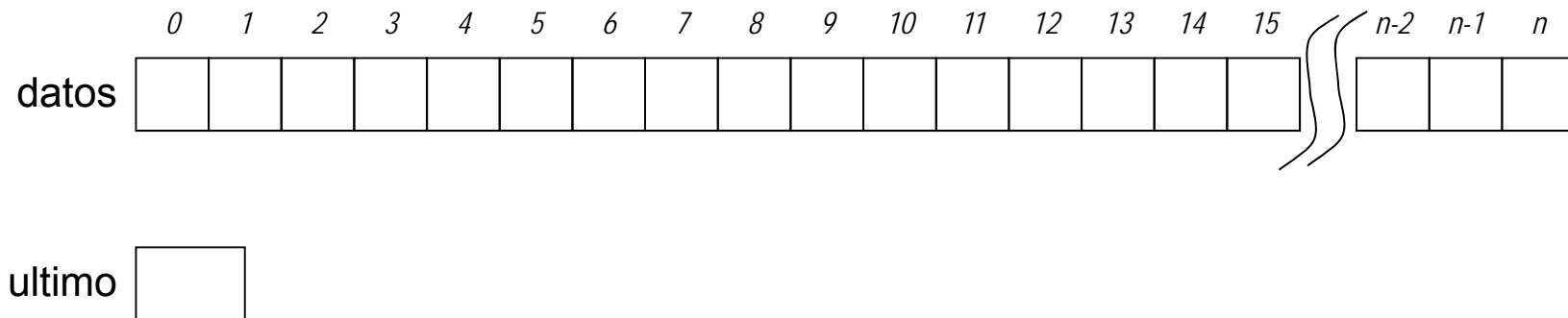
mientras $i < último$

$datos[i] = datos[i + 1]$

$i = i + 1$

fin mientras

$último = último - 1$



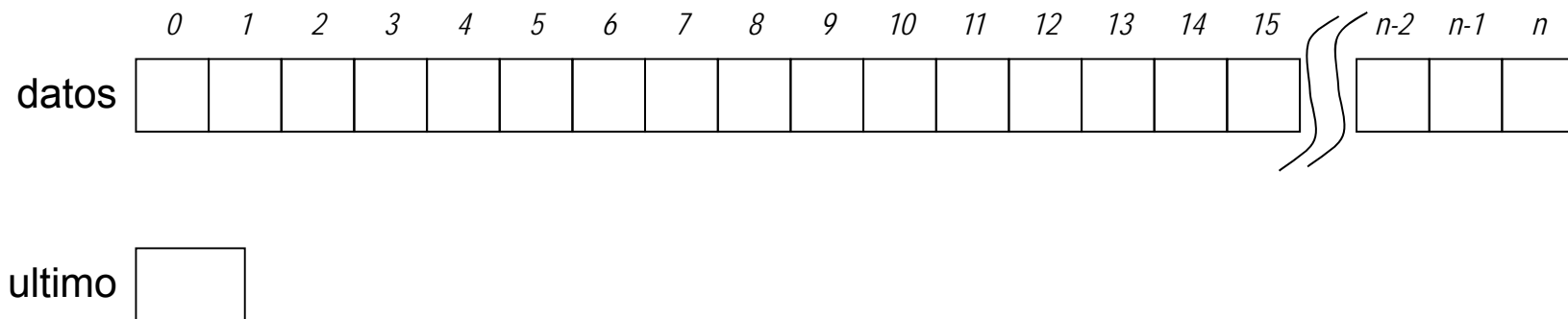
función: **primero**
 recibe: *lista(datos[], último)*
 regresa: posición

¿**vacía**(*lista*)?
 Sí: regresar: -1
 No: regresar: 0

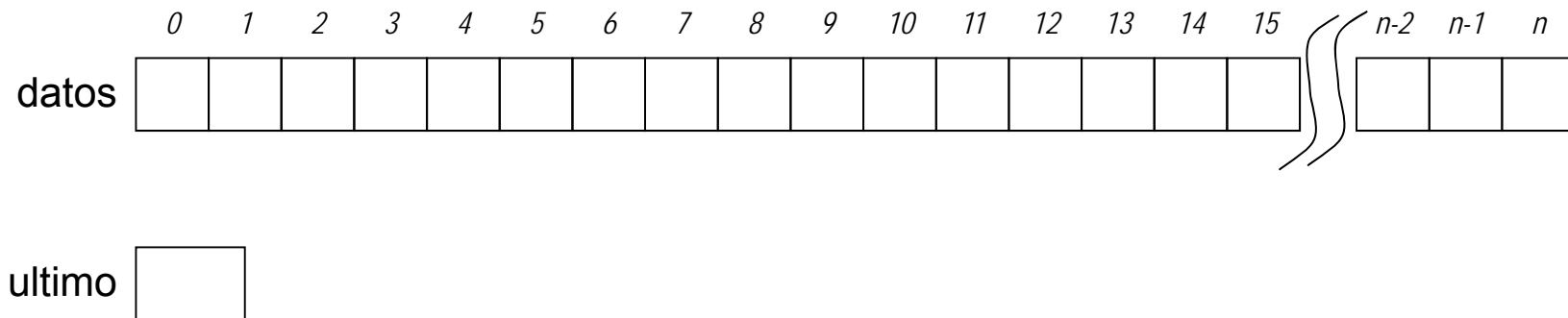
función: **primero**
recibe: *lista(datos[], último)*
regresa: posición

¿**vacía**(*lista*)?
Sí: regresar: -1
No: regresar: 0

```
tipo_pos primero (tipo_lista *L) {  
    if (vacía (L))  
        return -1;  
    else  
        return 0;  
}
```



función: **último**
 recibe: *lista(datos[], último)*
 regresa: posición
 regresar: *último*



función: **anterior**

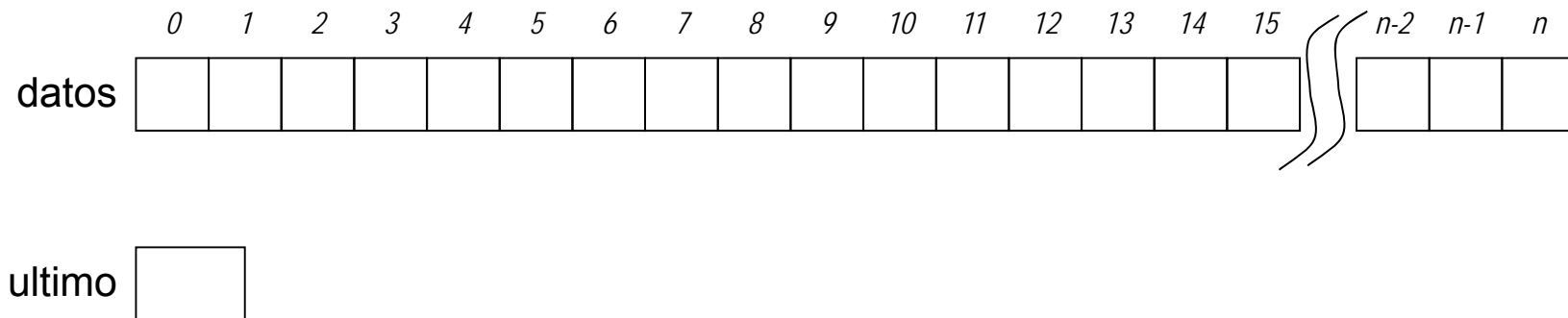
recibe: *pos*, *lista(datos[], último)*

regresa: posición

¿**vacía**(*lista*) ó $pos < 1$ ó $pos > ultimo$?

Sí: regresar: -1

No: regresar: $pos - 1$



función: **siguiente**

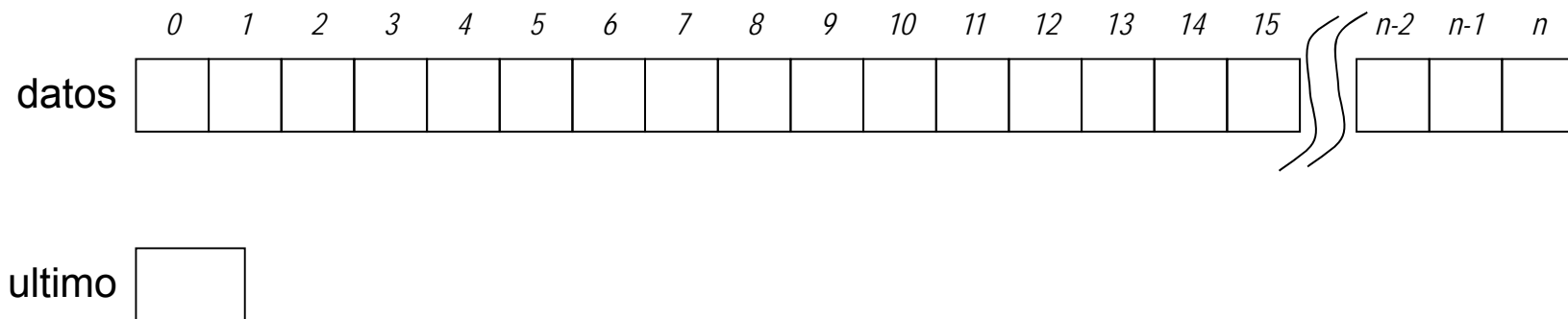
recibe: pos , $lista(datos[], \acute{u}ltimo)$

regresa: posición

¿**vacía**($lista$) ó $pos < 0$ ó $pos > \acute{u}ltimo - 1$?

Sí: regresar: -1

No: regresar: $pos + 1$

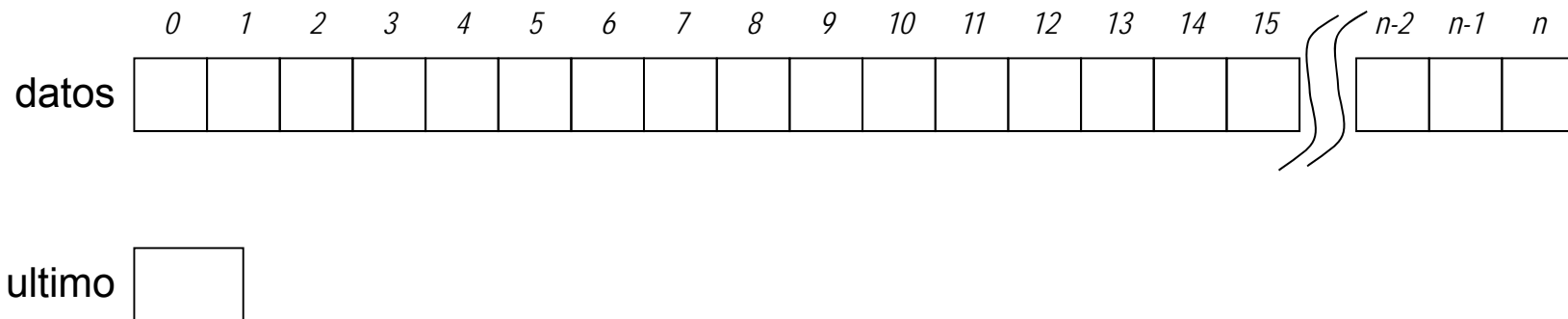


función: **localiza**

recibe: *elem*, *lista(datos[], último)*

regresa: posición

regresar: llamar: **búsqueda lineal**(*elem*, *datos*, *último*)



función: **recupera**

recibe *pos*, *lista(datos[], último)*

regresa: elemento

¿**vacía**(*lista*) ó $pos < 0$ ó $pos > último$?

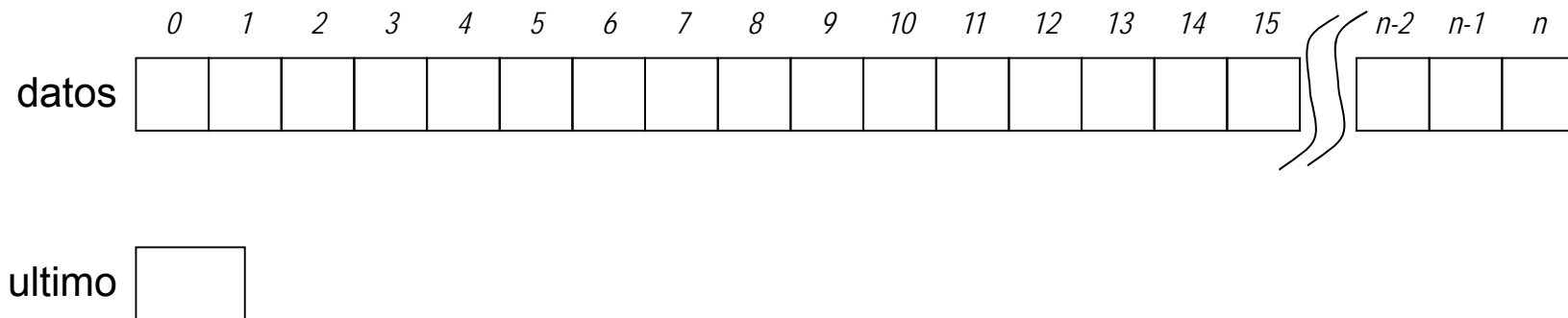
Sí: ¡Error de excepción!

Insuficiencia de datos

Terminar



No: regresar: *datos[pos]*

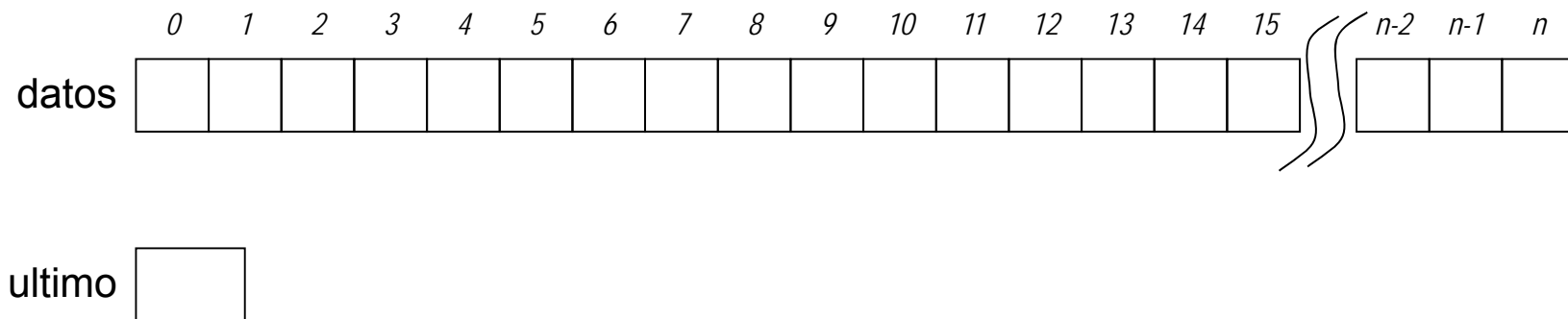


función: **ordena**

recibe: *lista(datos[], último)*

regresa: nada

llamar: **quicksort**(*datos*, 0 , *último*)

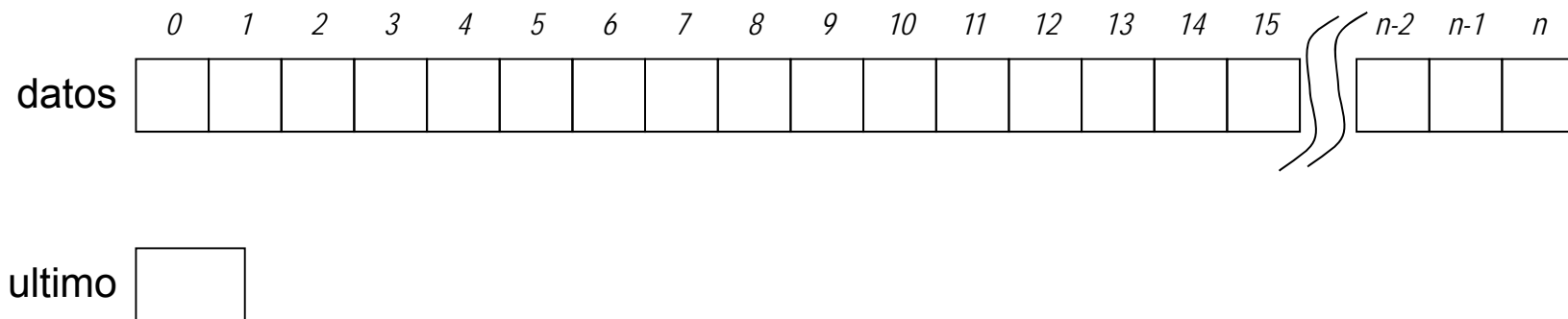


función: **imprime**
recibe: *lista(datos[], último)*
regresa: nada

$i = 0$
mientras $i \leq \text{último}$
 imprimir: *datos[i]*

$i = i + 1$

fin mientras



función: **anula**
recibe: *lista(datos[], último)*
regresa: nada
último = -1