Семинар №12

ФАКИ 2015

Бирюков В. А.

November 25, 2016

Структуры данных

Структуры данных

- Структура данных (англ. data structure) — определённый способ организации данных, так, чтобы их можно было использовать эффективно.
- Для разных задач более эффективными будут разные структуры данных.

Массив

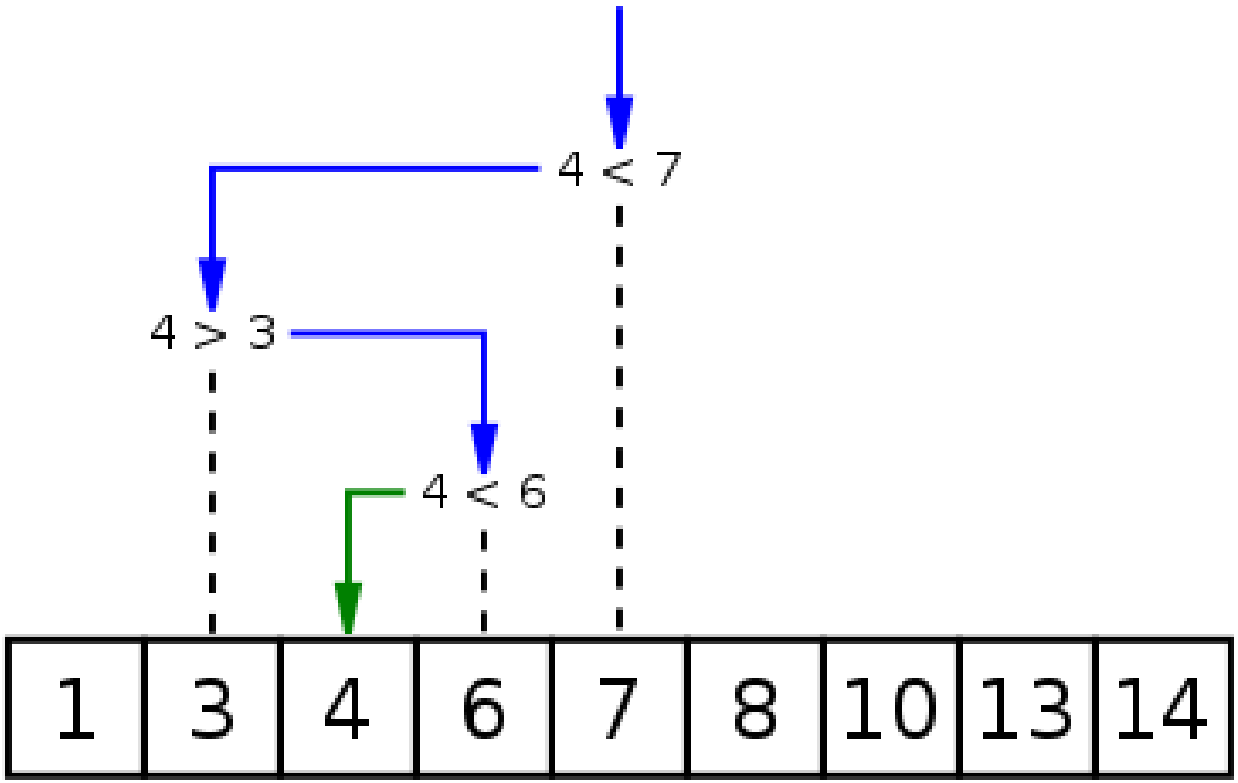
- Простейшая структура данных
- Статический или динамический
- Базовые операции:
  1. index – доступ к элементу
  2. insert – добавить элемент
  3. remove – удалить элемент
  4. find – найти элемент

Массив

	Массив
index	O(1)
insert	O(1)
remove	O(N)
find	O(N)

Упорядоченный массив

Бинарный поиск



Упорядоченный массив

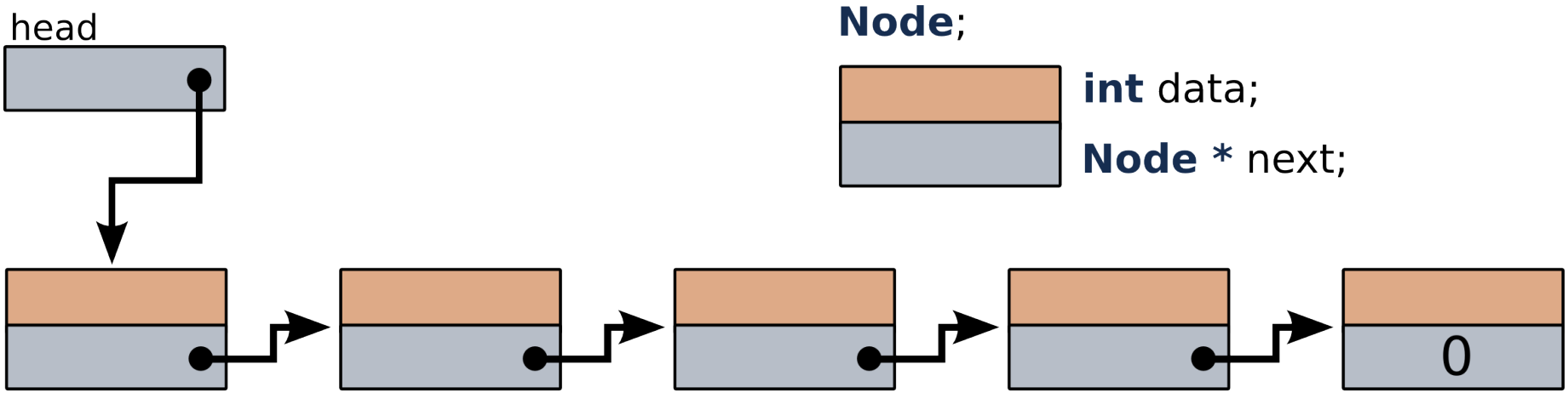
	Массив	Упорядоченный массив
index	O(1)	O(1)
insert	O(1)	O(N)
remove	O(N)	O(N)
find	O(N)	O(log(N))

Связный список

Связный список

```
struct Node {
  int data;
  struct Node * next;
};
struct Node * head;
```

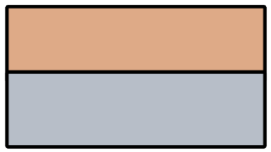
Связный список



Связный список

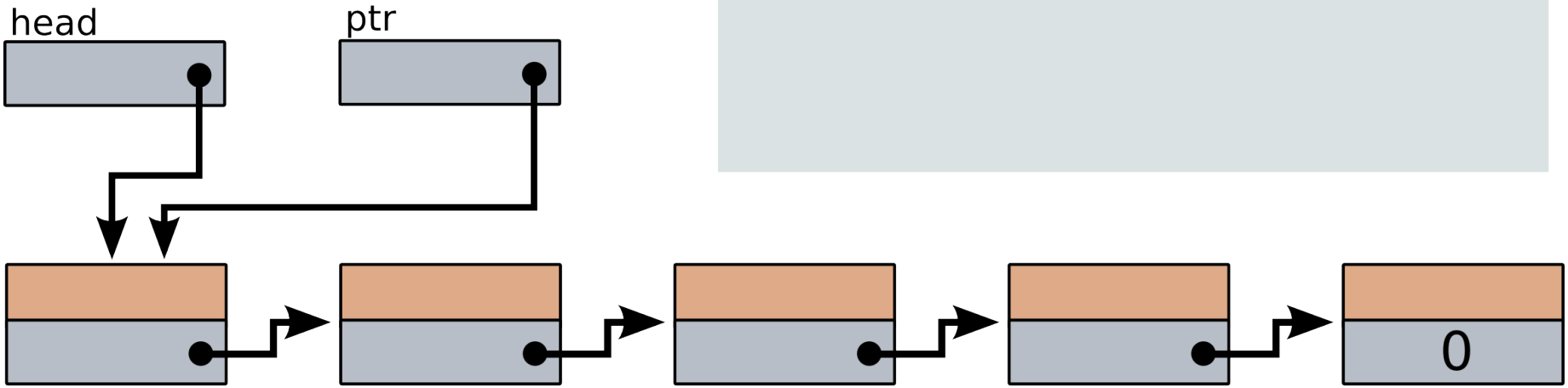
Обход связного списка - 1

Node;



int data;

Node * next;



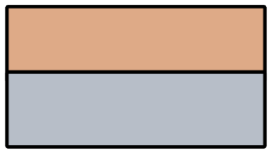
Код:

```
ptr = head;
```

Связный список

Обход связного списка - 2

Node;

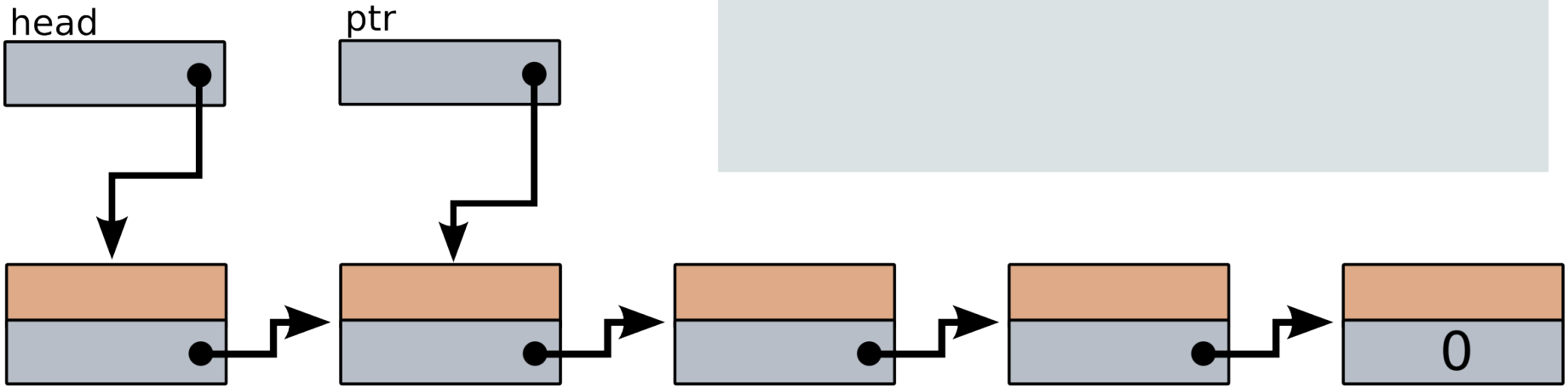


int data;

Node * next;

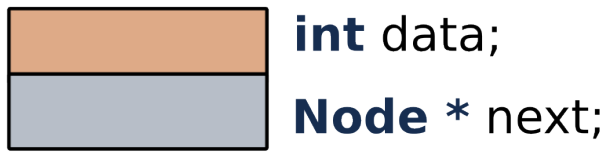
Код:

```
ptr = head;
ptr = ptr->next;
```



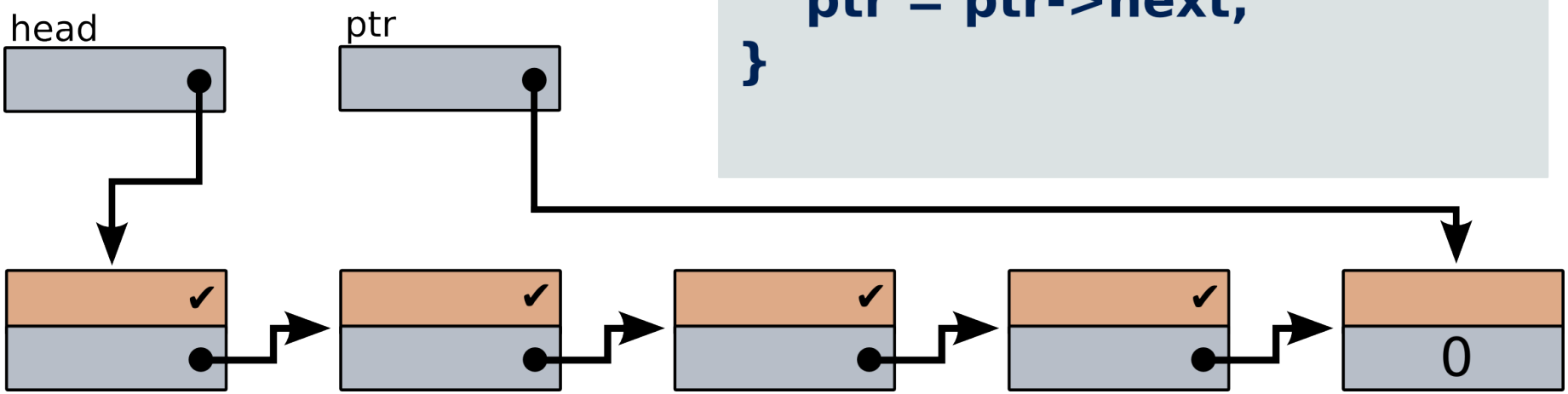
Связный список

Обход связного списка - 3



Код:

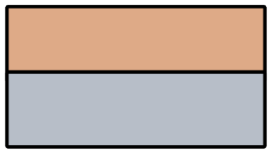
```
ptr = head;
while ( ptr->next != 0 ) {
    printf ("%d ", ptr->data);
    ptr = ptr->next;
}
```



Связный список

Удаление элемента списка - 1

Node;

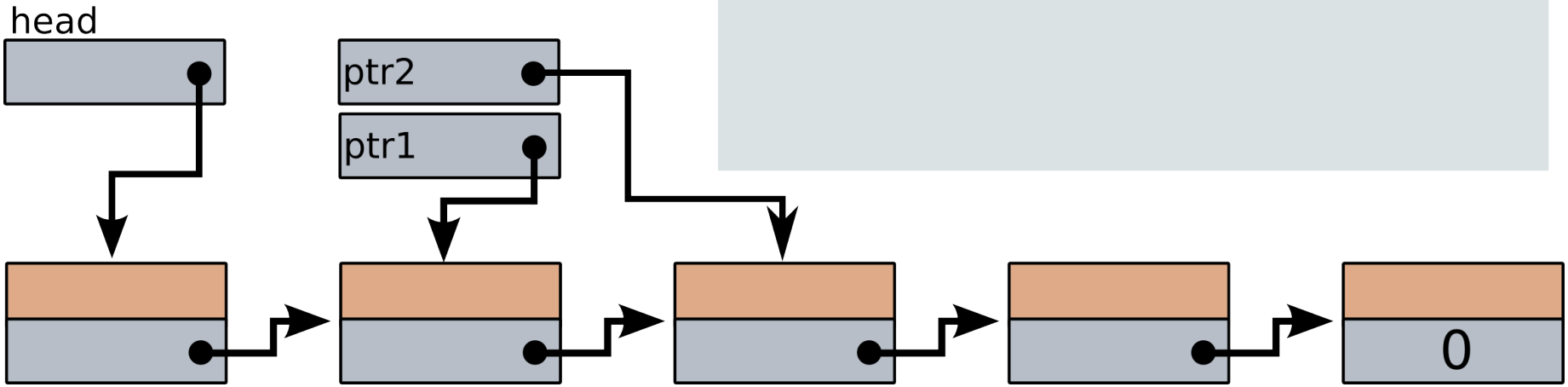


int data;

Node * next;

Код: Удаление n-го элемента

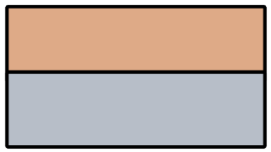
```
ptr1 = head->next;
ptr2 = ptr->next;
```



Связный список

Удаление элемента списка - 2

Node;

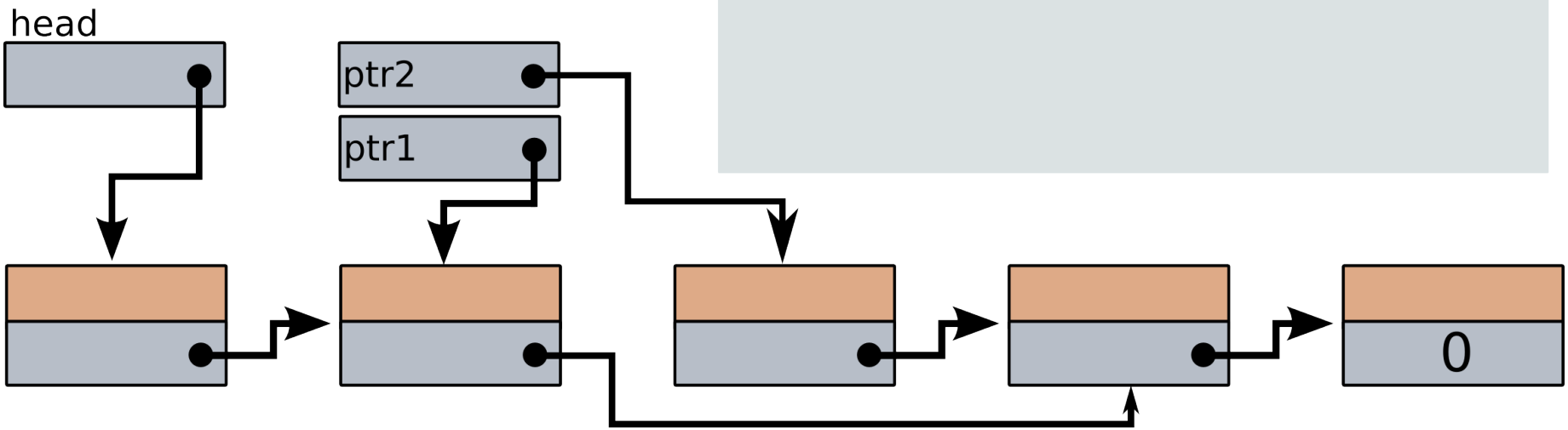


int data;

Node * next;

Код: Удаление n-го элемента

```
ptr1 = head->next;
ptr2 = ptr->next;
ptr1->next = ptr2->next;
```

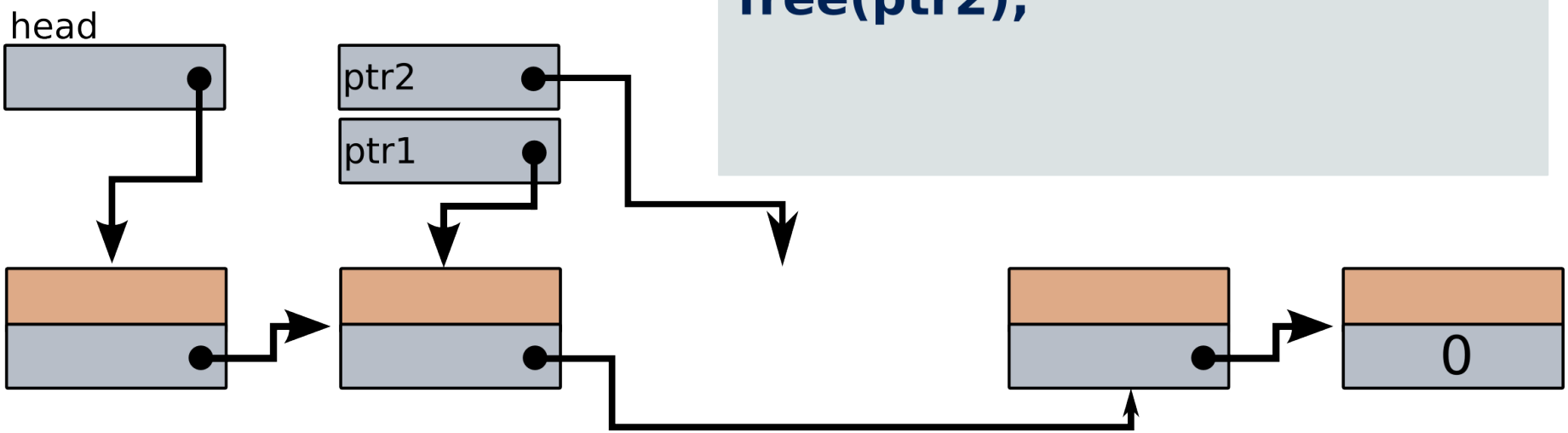


Связный список

Удаление элемента списка - 3

Код: Удаление n-го элемента

```
ptr1 = head->next;
ptr2 = ptr->next;
ptr1->next = ptr2->next;
free(ptr2);
```



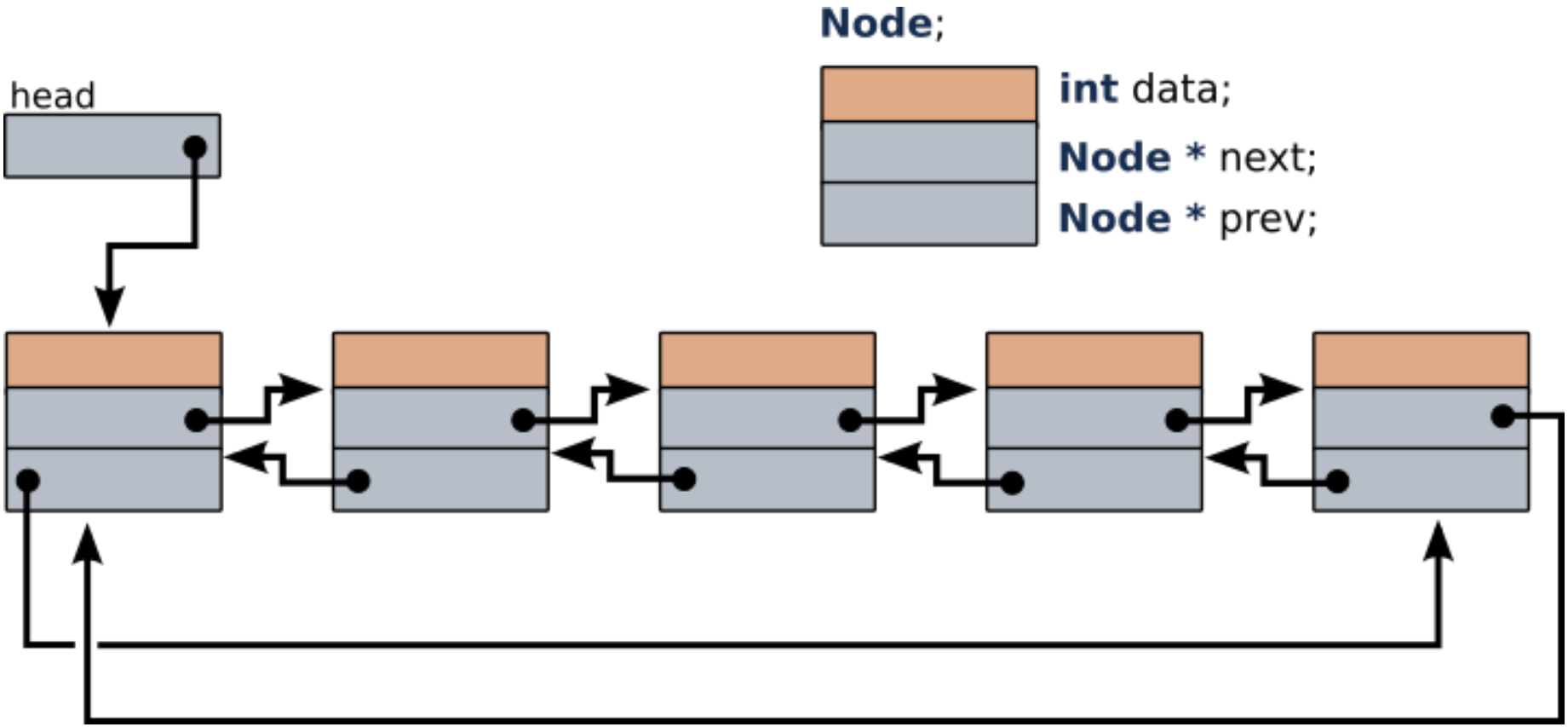
Связный список

- Простейшая структура данных
- Динамический
- Базовые операции:
  1. index – доступ к элементу
  2. insertToFront – добавить элемент в начало
  3. insertToBack – добавить элемент в конец
  4. insertAfter – добавить элемент после данного
  5. insertBefore – добавить элемент перед данным
  6. remove – удалить Известный элемент
  7. find – найти элемент

Связный список

	Список
index	O(N)
insertToFront	O(1)
insertToBack	O(N)
insertAfter	O(1)
insertBefore	O(N)
remove	O(1)
find	O(N)

Двусвязный список



Двусвязный список

	Список	Двусвязный список
index	O(N)	O(N)
insertToFront	O(1)	O(1)
insertToBack	O(N)	O(1)
insertAfter	O(1)	O(1)
insertBefore	O(N)	O(1)
remove	O(1)	O(1)
find	O(N)	O(N)

Valgrind

Задание

Задание

- Контест на списки