

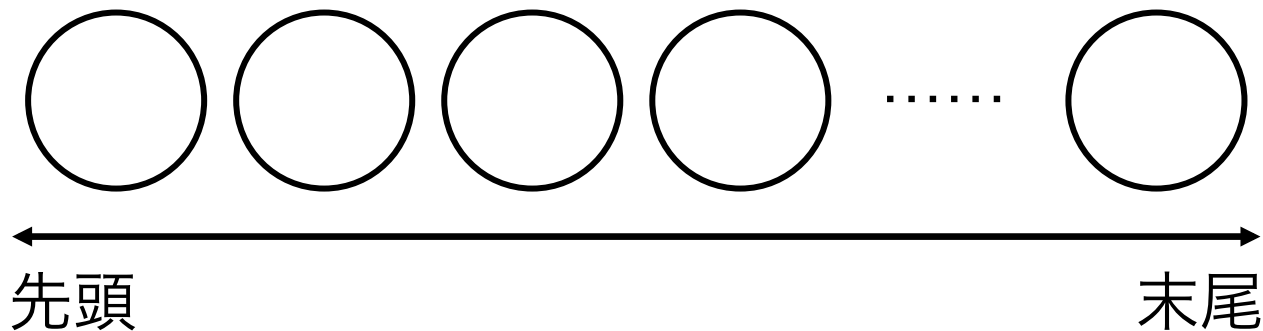
リスト (1)

クラスライブラリ応用

斉藤 (裕)

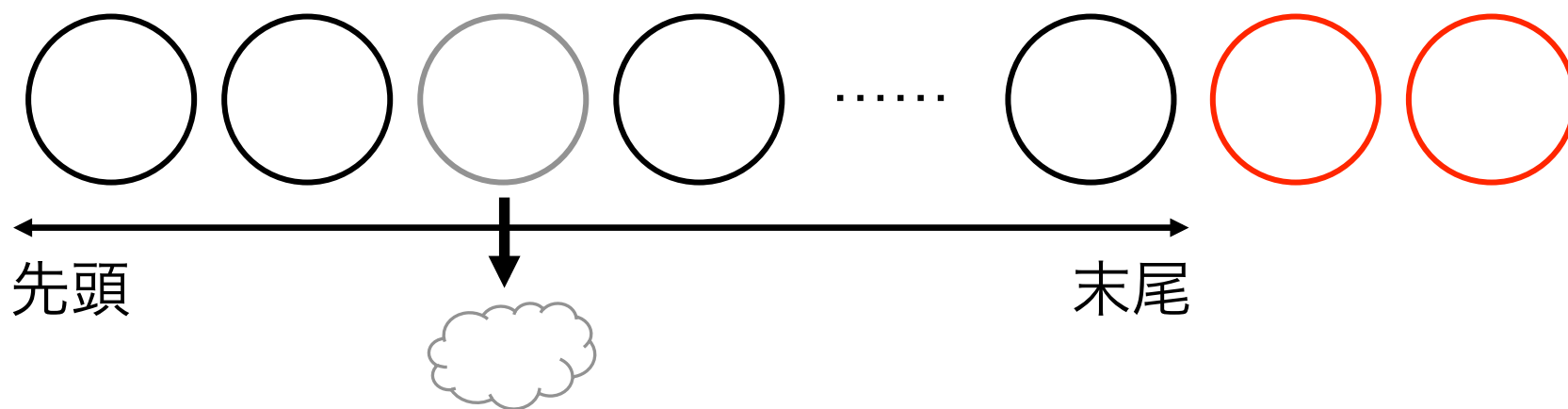
リストとは

- ・ 同じ種類のデータが1列に並んだ構造
- ・ 並びの順番にも意味がある
- ・ 教科書pp.280～



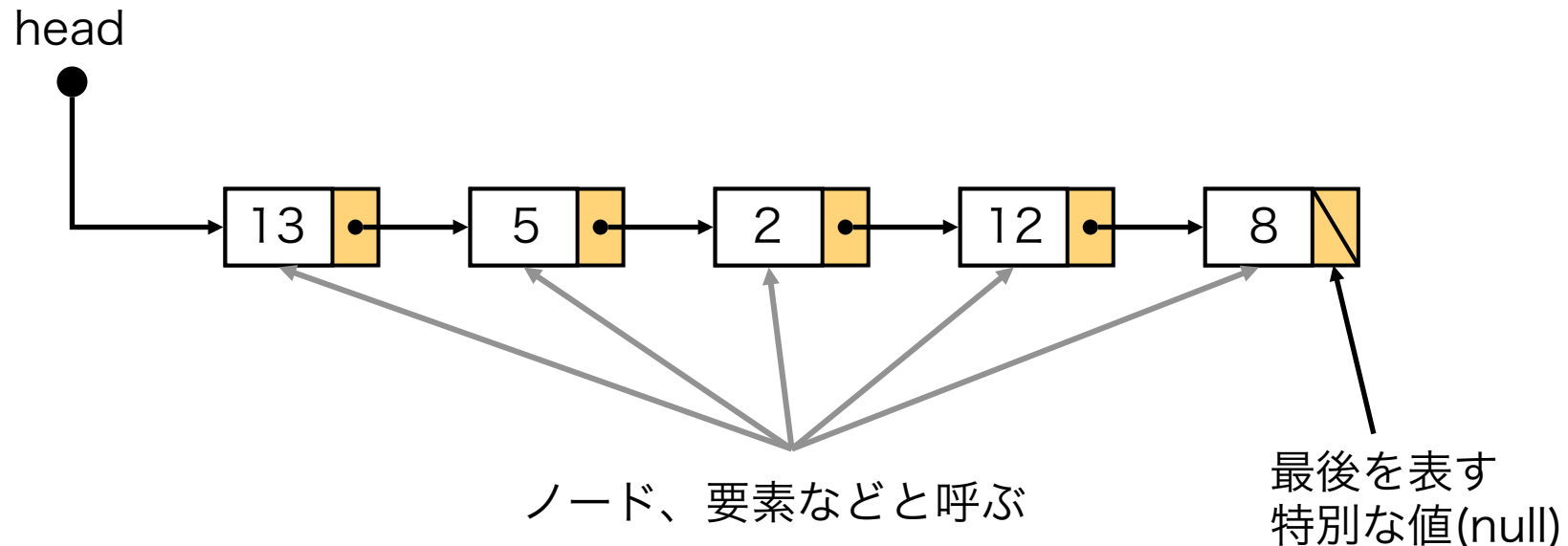
リストに対する操作

- ・ 追加
- ・ 削除
- ・ 中身の参照
- ・ 検索、他

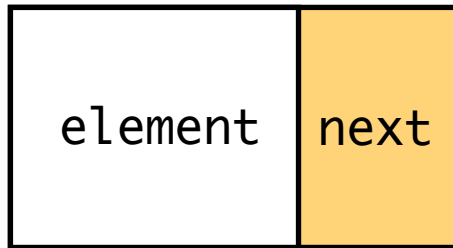


リストの表現方法

- ・ データを入れた箱を順につなぐ
 - 線形リスト (linear list) または単にリスト (list, linked list) と呼ぶ



ノードの表現方法

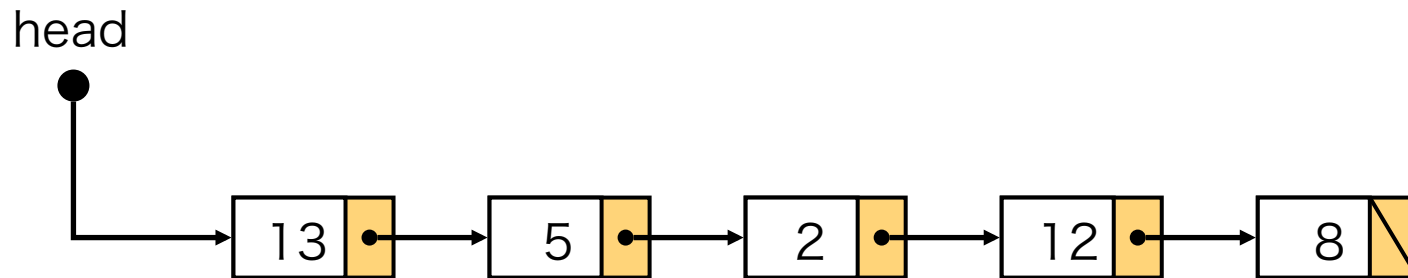


```
class Node {  
    int element;  
    Node next;  
}
```

- ・ ノードの情報
 - データ (今回は簡単化のため整数や文字列とする)
 - 次のノードへの参照 (参照=「在りか」に関する情報)

リストの使い方

- ・ 先頭のノードを管理する変数を用意
- ・ 目的のノードまで先頭からたどる
- ・ ランダムアクセスの機能はない




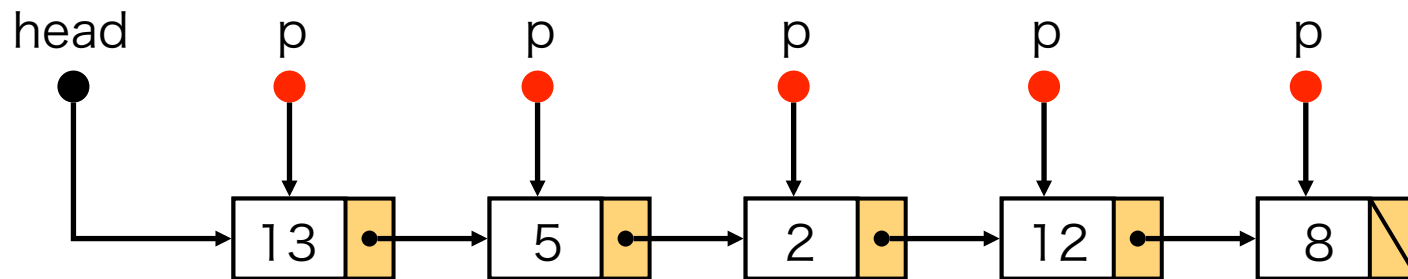
演習 1

- ・ 67, 41, 11, 24の順に並べられたリスト構造を図示しなさい。
headが先頭のノードを指し、リストの末尾には null (図では「/」で表す) を置くものとする。

リストの要素を順にアクセス

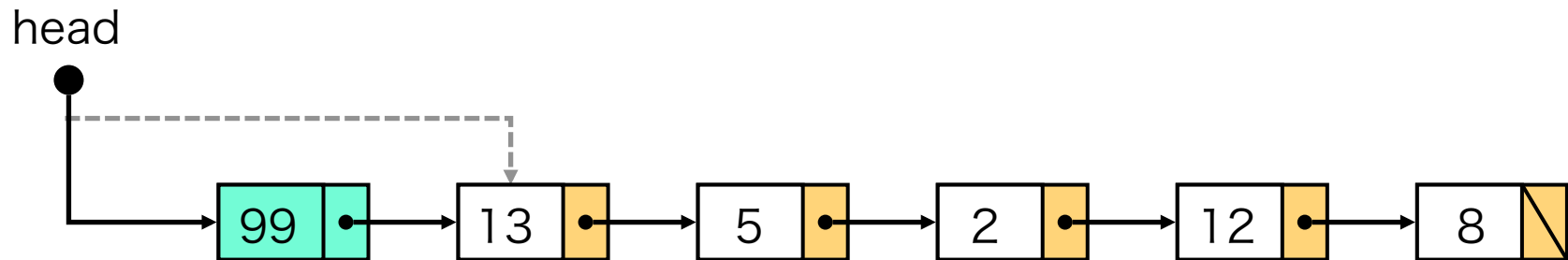
1. $p \leftarrow$ 先頭のノード
2. p が終点でない間 {
 - i. p のデータを参照
 - ii. $p \leftarrow p$ の次ノード
3. }

```
p ← head;   
while (p != null) {  
    p のデータを参照;  
    p ← p.next;  
}
```



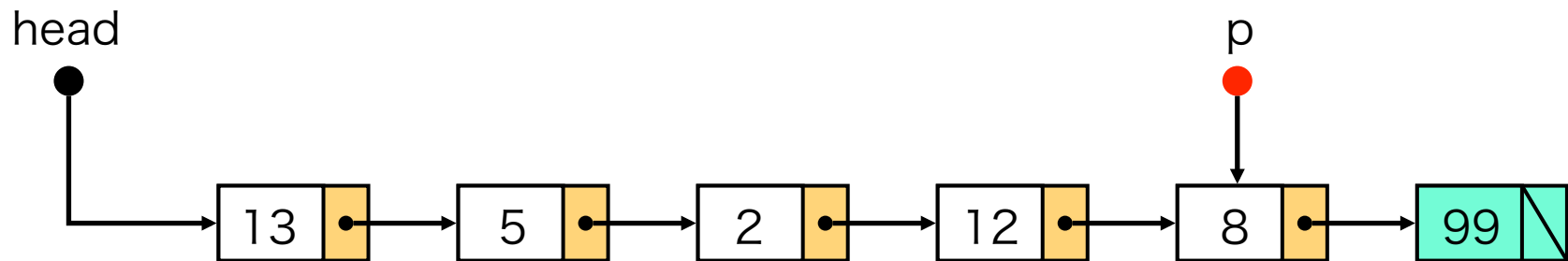
リストへのノードの追加(1)

1. リストの先頭(head)を新ノードとする
2. 新ノードの「次のノード」を元々の先頭ノードとする



リストへのノードの追加(2)

1. pの後ろに新ノードを追加
2. 新ノードの「次のノード」はnullとする



演習 2

1. 演習 1 のリストの末尾に新たなノード「28」を加えたときのリスト全体を図示しなさい。
2. リストの先頭ノードと、まだリストに追加していない新たなノードが与えられたとき、リストの末尾にノードを追加する手順を文章で記述しなさい。ステップごとに箇条書きで書くこと。
(例: 「1. リストを先頭から順にたどり末尾のノードを探し、pと置く」)

3. リストの内容を先頭から順に表示する手順は、以下の疑似プログラムで表せる。

```
p ← head
while (p != null) {
    p.elementを表示
    p ← p.next
}
```

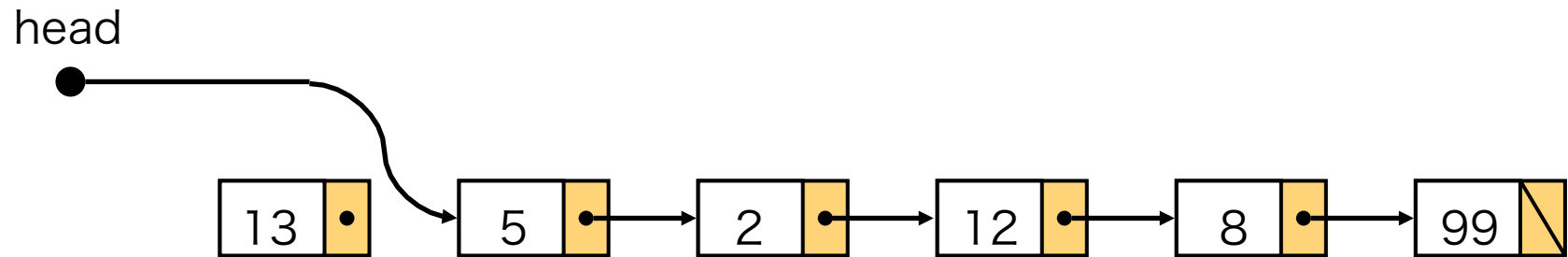
この例を参考に、変数headがリストの先頭ノードを指し、変数newNodeがまだリストに追加していない新たなノードを指しているとき、リストの末尾にnewNodeを追加する疑似プログラムを記述しなさい。

なお、追加を行うリストは空ではない (1つ以上のノードがすでに存在する) とする。
(もし、リストが空であった場合、リストの先頭に要素を追加する方法を使う)

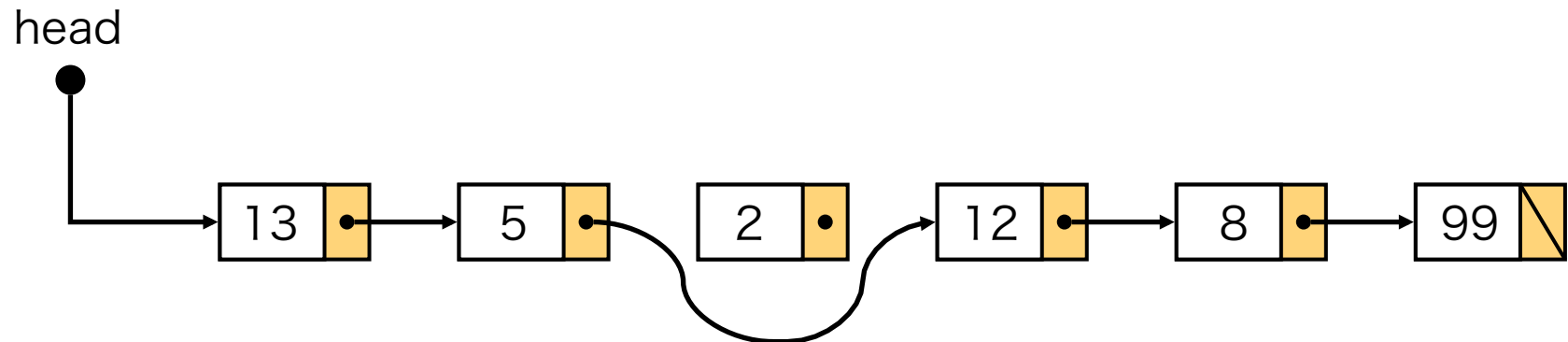
4. リストの図と疑似プログラムを用いて、正しくノードの追加ができることを、周りの友人同士でお互いに説明して確認しあいなさい。友人が見たらなければ副手と呼ぶこと。

リストからノードの削除

1. 先頭のノードを削除する場合

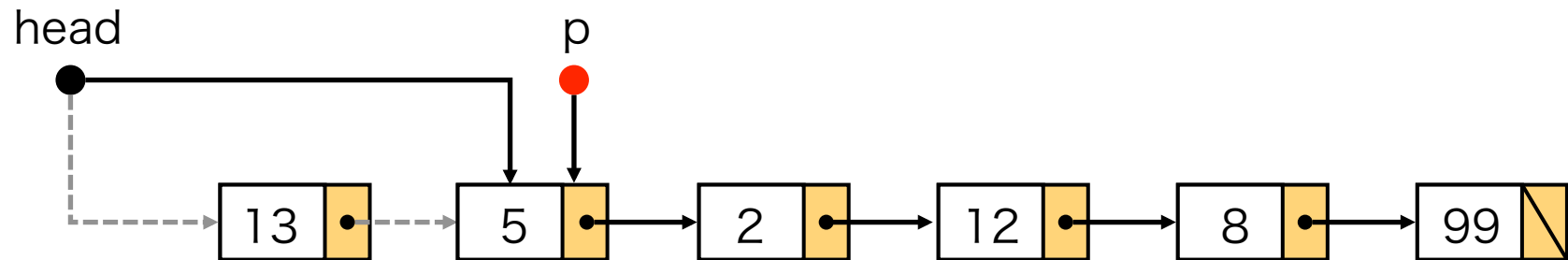


2. 先頭以外のノードを削除する場合



先頭のノードを削除

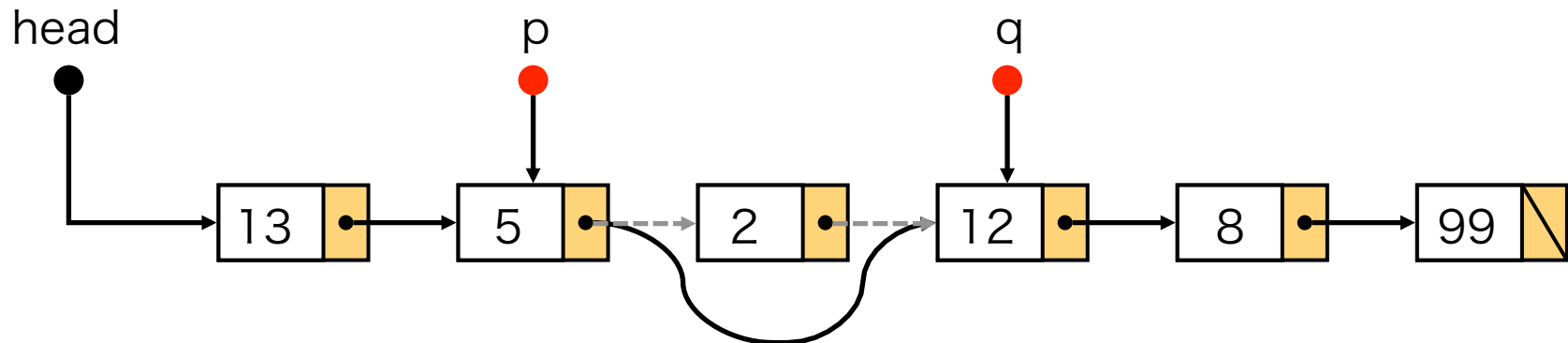
1. $p \leftarrow \text{headの次のノード}$
2. $\text{head} \leftarrow p$



先頭以外のノードを削除

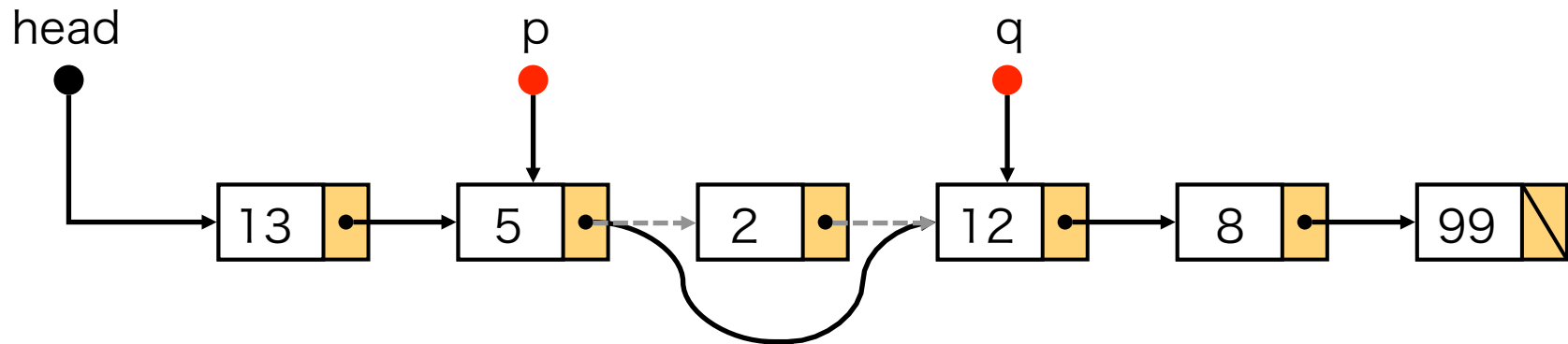
1. $p \leftarrow$ 削除ノードの手前のノード
2. $q \leftarrow$ 削除ノードの次のノード
3.

?



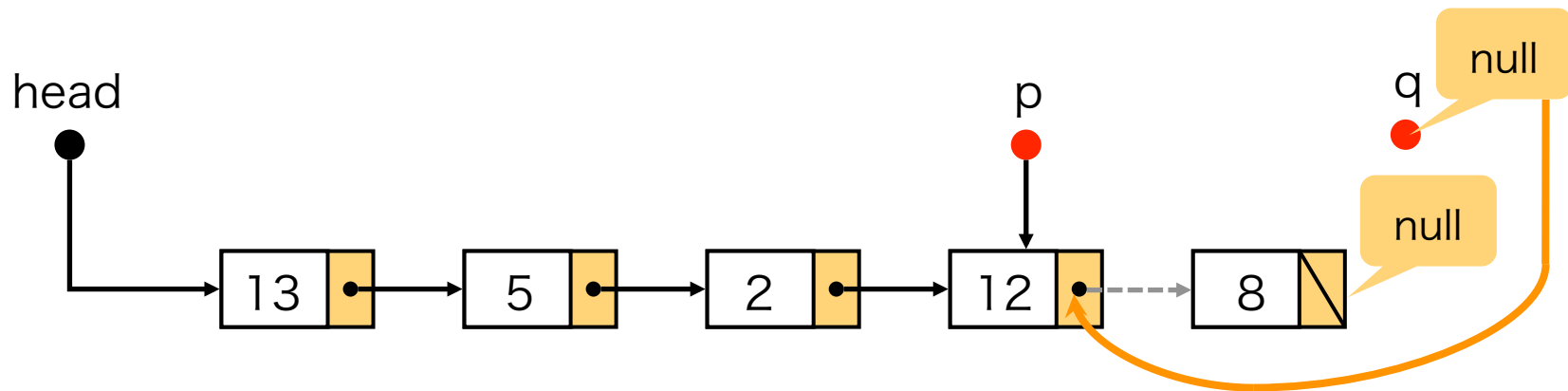
先頭以外のノードを削除

1. $p \leftarrow$ 削除ノードの手前のノード
2. $q \leftarrow$ 削除ノードの次のノード
3. $p.next \leftarrow q$



末尾ノードの削除も同様

1. $p \leftarrow$ 削除ノードの手前のノード
2. $q \leftarrow$ 削除ノードの次、すなわちnull
3. $p.next \leftarrow q$



配列とリストの比較

	配列	リスト
ランダムアクセス	○	×
データの挿入や削除	×	○
データの個数の変更	×	○

演習 3

1. 演習2のリストから41を削除することを考える。削除したときのリスト全体を図示しなさい。
2. リストの先頭ノードと、削除したい値が与えられたとき、リストからこの値を削除する手順を文章で記述しなさい。箇条書きで書くこと。リスト内にはデータの重複はないものとする。なお、削除を行うノードは先頭ではないと仮定する。
(例: 「1. リストを先頭から順にたどり、削除したい値を有するノードを探し、pと置く」)
3. 変数headがリストの先頭ノードを指し、変数vが削除したい値であるとする。リストからvを有するノードを削除する疑似プログラムを記述しなさい。
4. リストの図と疑似プログラムを用いて、正しくノードの削除ができることを、周りの友人同士でお互いに説明して確認しあいなさい。友人が見たらなければ副手を呼ぶこと。