



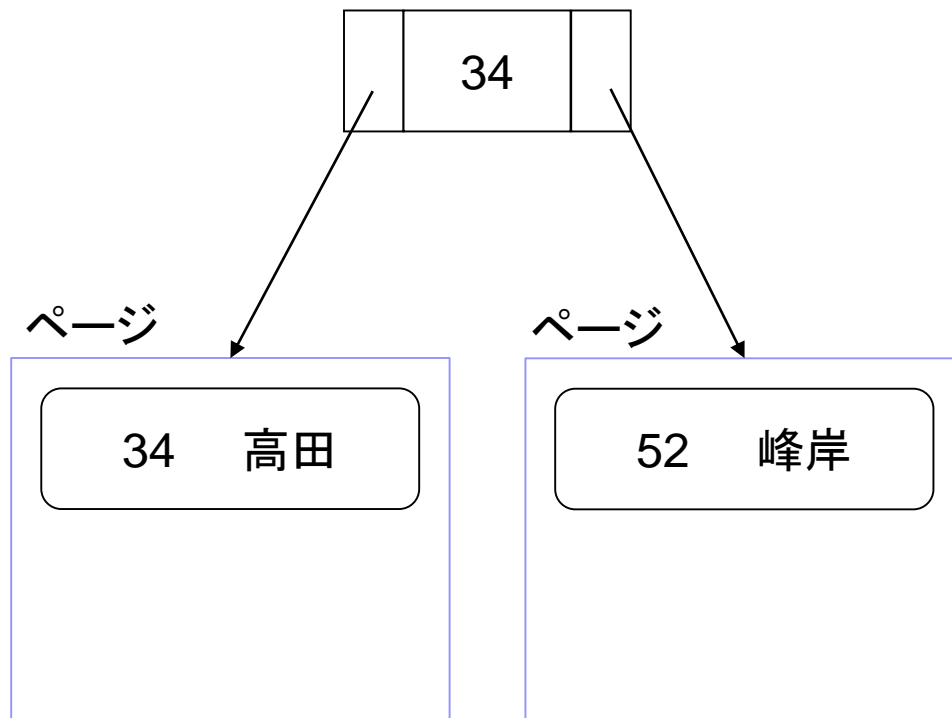
データベース (第11回・演習問題編)

情報工学科 木村昌臣



ISAM

1.ISAMを使ったインデックス (初期状態・追加1)

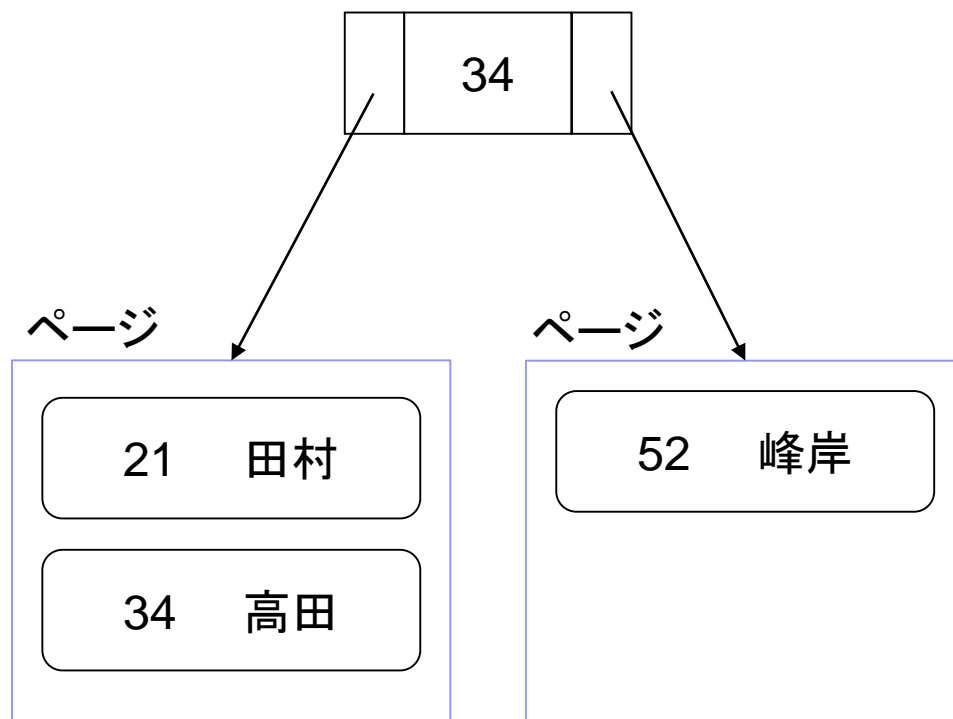


1ページに2レコード入ることを前提とする
また、簡単のため、ISAMオーダーは2とする。

21 田村

を追加したい

2.ISAMを使ったインデックス (初期状態・追加2)



23 中村

を追加したい

3.ISAMを使ったインデックス (追加3)

2.で得られた木に

31

森久保


を追加

4.ISAMを使ったインデックス (追加4)

さらに

24 佐藤

を追加



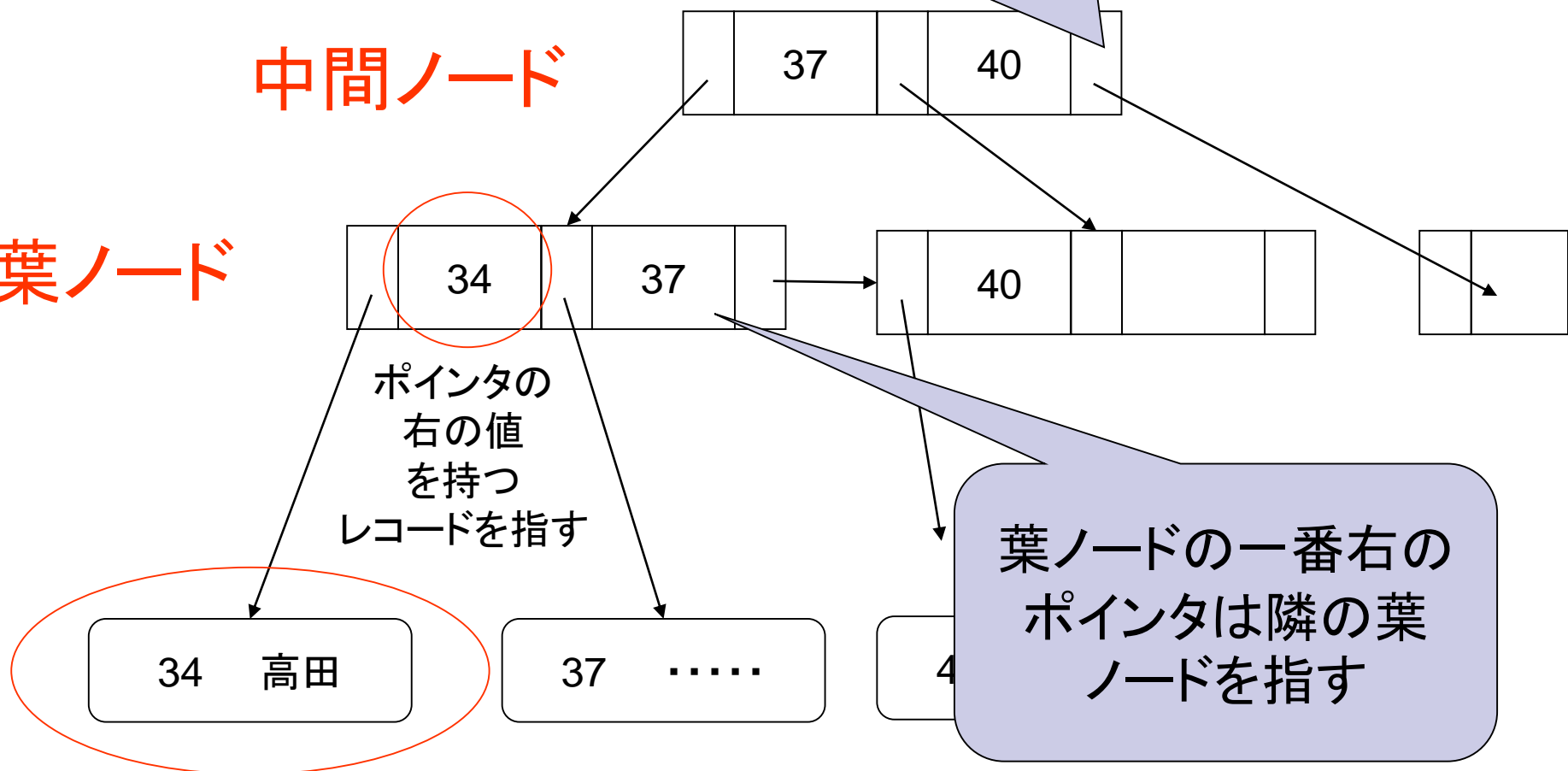
B+木

B+木のノード

中間ノードのポインタは**全ての子ノードを指す**(ただし、空
きの場合もある)

中間ノード

葉ノード

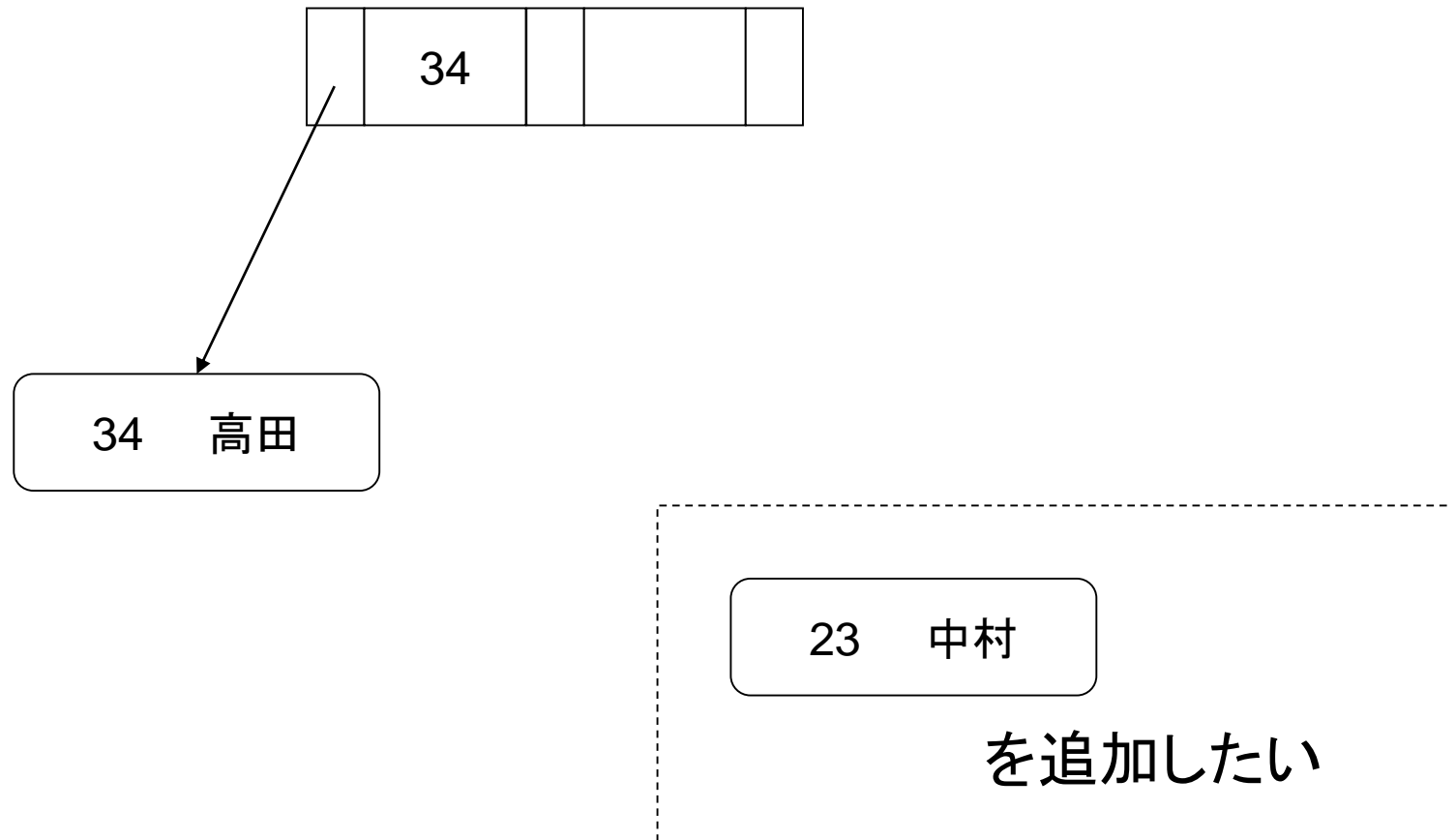


※通常、1ノードが1ページ(=ストレージからのデータ読み込みの最小単位)に対応

B+木へのレコード挿入の方法

1. レコードが入るべき葉ノードを発見
 - 葉ノードに空きがあればそのノードからレコードへのポインタを張る
 - 左から埋める
2. 葉ノードに空きがない場合は、ノードを分割
 - 元のノードに、追加分を含めたレコード数の半分（つまり、入りうるレコード数+1の半分。ただし、小数点切り上げ）を残し、新しいノードに残りを移す
3. 分割されたノードの親ノードにそのノードの探索キーの最大値を挿入し、葉ノードへのポインタを更新
 - もし親ノードに空きがなければ親ノードを分割

1.B+木を使ったインデックス (初期状態・追加1)



2.B+木を使ったインデックス (追加2)

1.で得られたB+木に

31

森久保

を追加

ヒント: このままでは葉ノードからレコードを指す
ポインタが足りなくなるから、分割が必要。
葉ノードが指しうるレコードの数は二つだから、
追加分を含めると三つのレコードを指すことにな
る。元のノードにいくつ残せばよい？

3.B+木を使ったインデックス (追加3)

さらに

24	佐藤
----	----

 を追加

4.B+木を使ったインデックス (追加4)

さらに 25 吉岡 を追加

5.B+木を使ったインデックス (追加5)

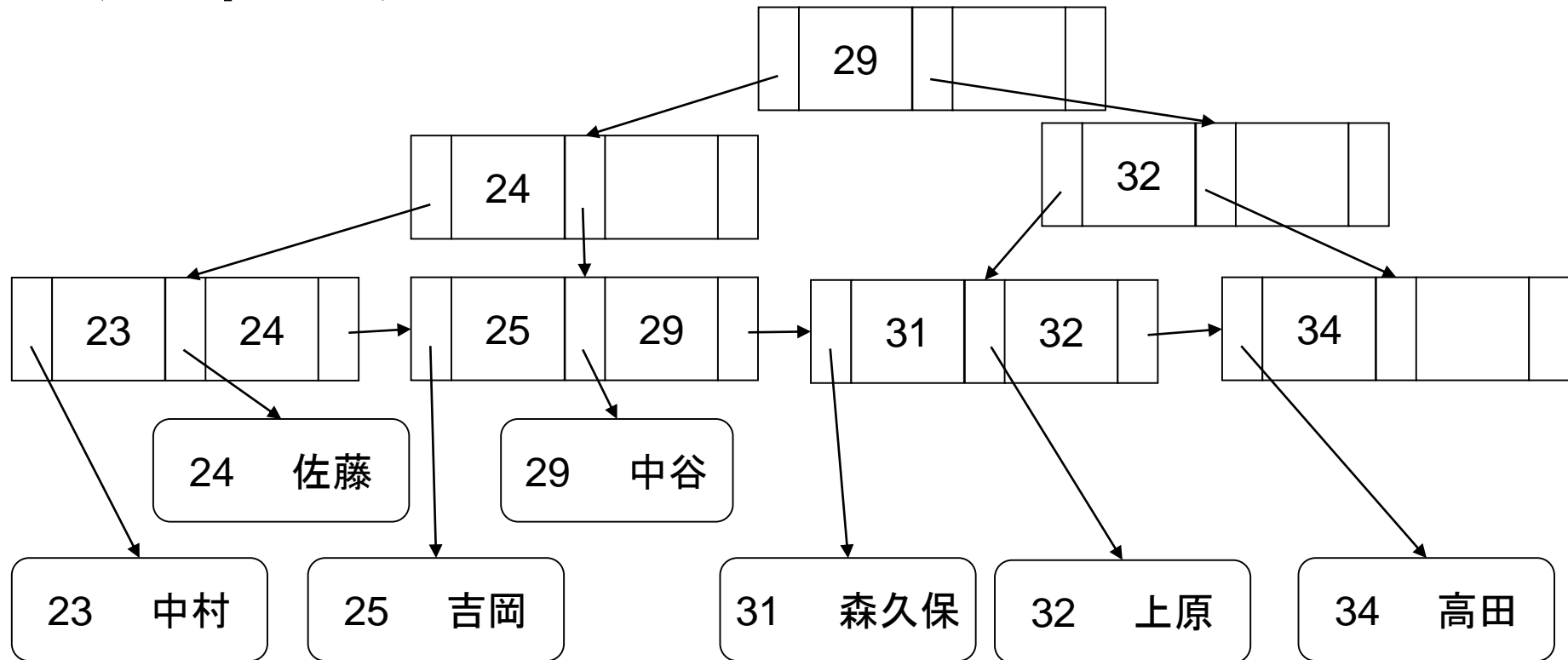
さらに

29

中谷

を追加

6.B+木を使ったインデックス (削除1)



24 佐藤

を削除したい

7.B+木を使ったインデックス (削除2)

さらに

32 上原

を削除

8.B+木を使ったインデックス (削除3)

さらに

31 森久保

を削除