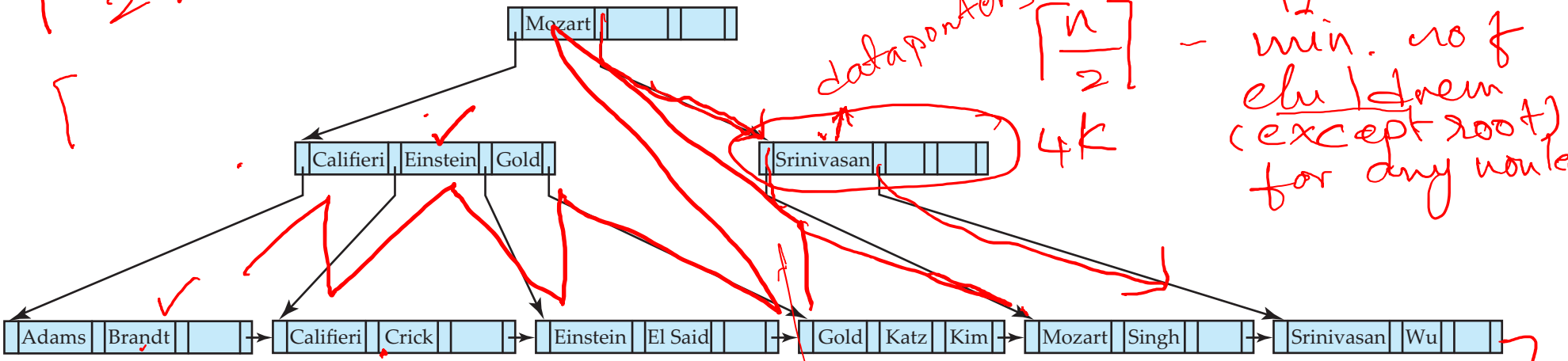
COL 362 & COL 632

Indexing

03 Mar 2023

Queries on B$^+$-Trees



Find record with search-key value *V*.

1. *C=root*
2. While C is not a leaf node {
   1. Let *i* be least value s.t. $V \leq K_i$.
   2. If no such exists, set *C* = *last non-null pointer in C*
   3. Else { if (V= K_i) Set C = P_{i+1} else set *C* = P_i}

   }
3. Let *i* be least value s.t. $K_i = V$
4. If there is such a value *i,* follow pointer P_i to the desired record.
5. Else no record with search-key value *V* exists.

Analysis of B$^+$-trees

- No. of search keys: 1,000,000
- Block size: 4K
- Size of an index entry: 40B
- Max. no. of search keys per block:
- Max. height of the tree:
- No. of block accesses per query:

$n = \frac{4K}{40B} \approx 100$

50

$\log_{50} 1 \times 10^6 = 4$

Key + Pointer

100

4

$(4 + 1)$

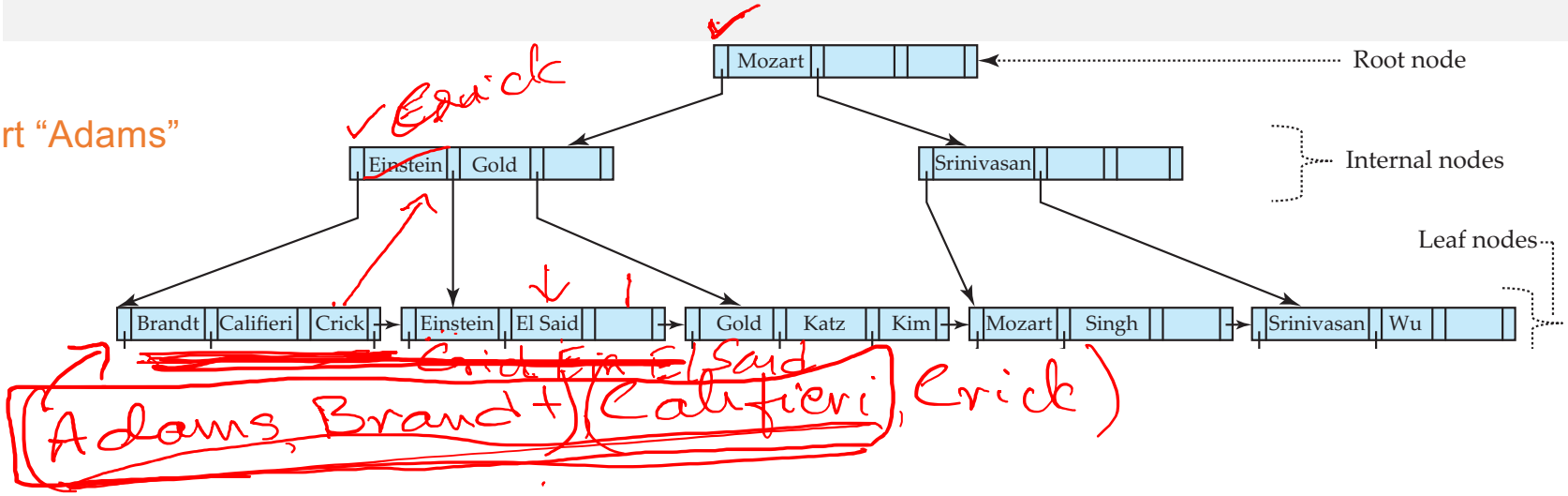
4

$\log_2 1 \times 10^6$

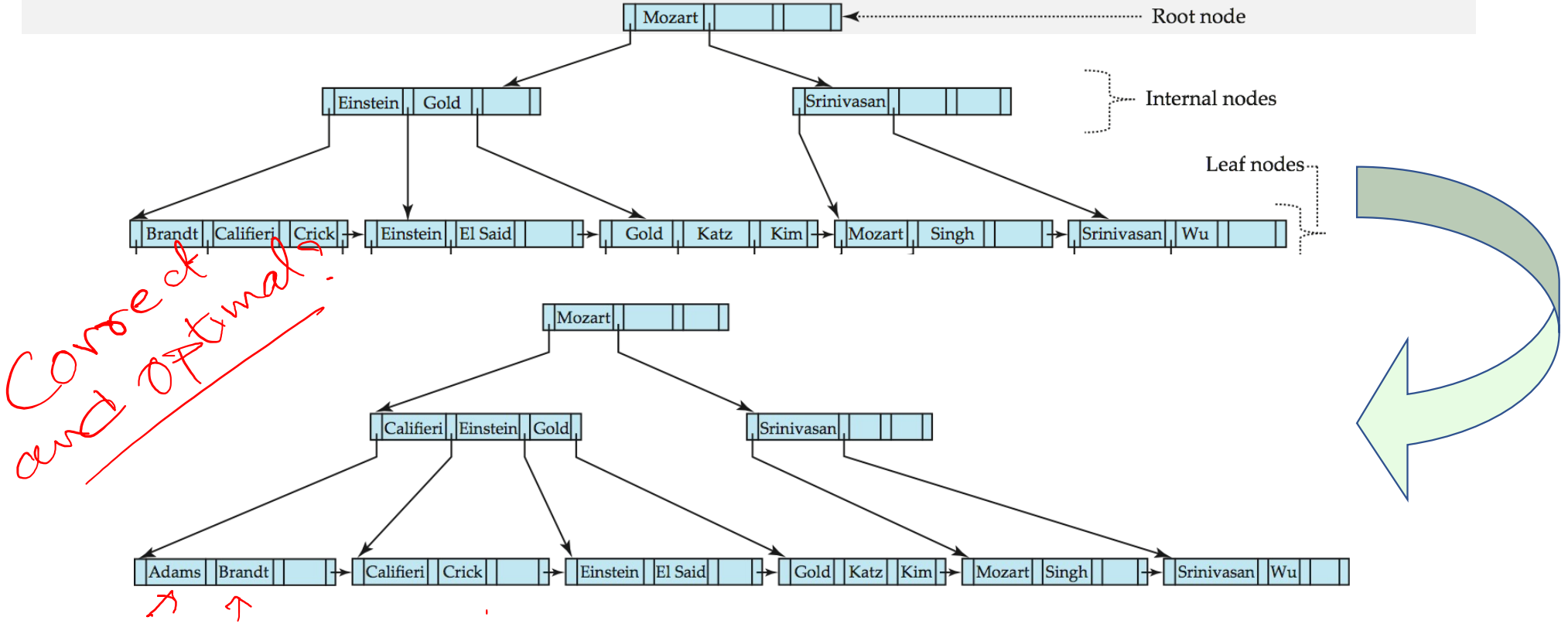
B+-trees – insertion

B+-Tree Insertion

Insert “Adams”



B+-Tree Insertion



B+-Tree before and after insertion of “Adams”