

CMPS 340 File Processing
Examples of insertion/deletion in a B⁺-tree

Consider the B⁺-tree T illustrated below. Let its B-tree —which is comprised of the internal nodes— have order four. Suppose that the leaves are large enough to hold up to four records. In order to ensure that space utilization is no worse than about 50%, we allow a minimum of two records in a leaf. (Note that each record contains two fields, the first being the ordering field (an integer) and the second being a character string.)

For each operation in the list **(a)** through **(i)**, show the B⁺-tree that results from performing that operation on T . Assume, in carrying out the operations, that *redistribution* is used whenever possible. That is, split an overflowing node only if all its adjacent siblings are full. Similarly, concatenate two nodes only if all adjacent siblings of the underflowing node are on the verge of underflowing. (*Note:* Redistribution is never to extend beyond two adjacent siblings!)

- | | | |
|--|--|--|
| (a) insert 56 ink (b) insert 11 ink (c) insert 31 ink | (d) insert 20 ink (e) delete 48 (f) delete 50 | (g) delete 61 (h) delete 78 (i) delete 96 |
|--|--|--|

