CS 240: Data Structures and Data Management Winter 2023

Tutorial 12: April 10

This tutorial handout covers problems from Text Compression (Module 10) and External Memory (Module 11). There are 3 problems in total – 2 easy [E] and 1 hard [H].

1. [E] Consider Burrows-Wheeler Transforms:

a) Encode the following string using BWT: `MISSISSIPPI`

b) Decode the following string using the inverse BWT: `AIMOEOOPN$TOA`

2. [E] For the following(2, 4)-Tree, perform the following operations:

a) Insert 30, Insert 75, Insert 24, Insert 56

b) Delete 56, Delete 24, Delete 75, Delete 30

When deciding between successor/predecessor, choose the successor. When deciding between left or right sibling for transfer/merge, select the right sibling.
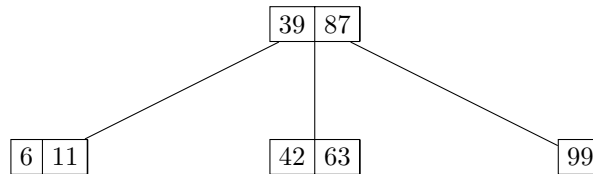
3. [H] Consider a modification to LZW to expand the dictionary faster: at every step, the encoder adds two new dictionary entries instead of one, when possible; one entry corresponds to the current string being encoded + the next character (like the usual LZW), while the other entry corresponds to the current string being encoded + the next two characters.

For example, if the text is `APPLE`, then after encoding `A` (as 65), the encoder adds two entries to the dictionary: `AP` at 128, and `APP` at 129. Note that after encoding `L` (as 76), the encoder only adds one new entry, `LE` at 134, and that no entries are added after encoding `E` (as 69).

a) Encode the following string with the modified LZW: `BAN_ANANAS_AND_BANANAS`.

b) Modify the LZW decoding algorithm to decode strings that were coded with this modified LZW.

c) Decode the following string that was encoded with the modified LZW:

$$82 - 79 - 84 - 79 - 95 - 77 - 69 - 138 - 78 - 133 - 147 - 128 - 143$$