Find actual subsequences? Quick ex!

ex 1: \( X = BCEAD \) \( m = 5 \), \( n = 3 \)
\( Y = ADA \)

1. Find LCS:

   \[
   \begin{array}{cccc}
   O & A & D & A \\
   0 & 0 & 0 & 0 \\
   B & 1 & 0 & 0 \\
   C & 0 & 0 & 0 \\
   E & 0 & 0 & 0 \\
   A & 0 & 1 & 1 \\
   D & 0 & 1 & 2 \\
   \end{array}
   \]

   \(\# \) Note: First A's in \( X, Y \) matched

   \(\#\) although "wrong" A's matched later

   \(\#\) where we correctly decide that matching the first A in \( X \) yields longer LCS

   \(\#\) look at arrows:

   \[
   \text{see DP formulation}
   \]

   How to find actual LCS?

   - Keep arrows that indicate where best LCS was from

   - Notice: Only indices with \( \uparrow \) indicate that corresponding char is in LCS. So store these chars.

   \[
   \begin{array}{cccc}
   O & A & D & A \\
   0 & - & - & - \\
   B & \leftarrow & \downarrow & \downarrow \\
   C & \leftarrow & \downarrow & \downarrow \\
   E & \leftarrow & \downarrow & \downarrow \\
   A & \leftarrow & \leftarrow & \leftarrow \rangle \text{A} \\
   D & \leftarrow & \leftarrow & \leftarrow \rangle \text{A} \\
   \end{array}
   \]

   Break ties between \( \leftarrow \), \( \uparrow \) by arbitrariness choosing \( \leftarrow \)

1. Start at \( C[n][j] \), \( \text{ex:} \leftarrow \rangle \text{A} \)

2. Follow arrows

3. If see "\( \text{A} \)" output character

4. LCS is reverse of outputted chain

LCS = AD
LCS Problem

another example

$X = \text{BCAAD}$, $m = 5$, $n = 4$

$Y = \text{AACD}$

```
    D   A   A   C   D
  D   0   1   0   0   0
B   0   0   0   0   0   0
C   0   0   0   0   0   0
  0   1   1   1   1   1
  0   2   2   2   2   2
  0   2   2   2   2   3
```

"incorrect" match

Find sequence:
Start at $(m, n)$
For every output character (in reverse)

$LCS(\overline{x}_5, \overline{y}_4) = \text{-AAD}$
Text Verification

Corrupted text file with spaces + punctuation removed

Goals:
1. Determine if the text is a valid string.
2. If so, reconstruct the string.

Assume: (1) Ignore case
2. Given a dictionary that takes as input string w:

\[ \text{Dict}(w) = \begin{cases} 
\text{true (1)} & \text{if } w \text{ is a valid word (including "a" and "I")} \\
\text{true (1)} & \text{if } w \text{ is empty string} \\
\text{false (0)} & \text{otherwise} 
\end{cases} \]

Note: S = "thinknot" : 2 correct solutions: think not thin knot

ex. S = thinnothing
1 2 3 4 5 6 7 8 9 0 1 2 3

Possible to solve in \( O(n^2) \), \( n = 151 \)