

# Explanation

The LCS approach requires the insight that:

- $\text{LCS}(s, \text{reverse}(s))$  gives you the longest palindromic subsequence
- This is because any common subsequence between a string and its reverse must be palindromic

## The Key Insight Visualized

Let's say we have string `s = "babad"` and its reverse `rev = "dabab"`

```
s = "b a b a d"
rev = "d a b a b"
```

Now, let's find what subsequences they have in common:

### Example 1: Common subsequence "aba"

- In `s`: pick positions 0, 2, 3 → "babad" → "aba"
- In `rev`: pick positions 1, 2, 4 → "dabab" → "aba"

Notice: "aba" is palindromic! ✓

### Example 2: Common subsequence "bab"

- In `s`: pick positions 0, 2, 4 → "babad" → "bab"
- In `rev`: pick positions 2, 4, 0 → "dabab" → "bab"

Notice: "bab" is palindromic! ✓

## Why This Always Works

Here's the crucial insight: **If a subsequence appears in both a string and its reverse, it must read the same forwards and backwards.**

Think about it step by step:

1. Let's say we have a common subsequence: `C1C2C3...Ck`

2. This subsequence appears in the original string `s` from left to right in some positions
3. The same subsequence also appears in the reversed string `rev`
4. But here's the key: in the reversed string, everything is flipped! So if `c1c2c3...ck` appears in the reversed string, it means the original string actually contained `ck...c3c2c1` in those positions when read left-to-right
5. For the same sequence to appear in both, we need:
  - `c1c2c3...ck` (in original)
  - `c1c2c3...ck` (in reversed, which means `ck...c3c2c1` was in original)
6. This is only possible if `c1c2c3...ck = ck...c3c2c1`, which means it's a palindrome!

## Another Way to Think About It

Imagine you're reading a subsequence from left to right in the original string. If that exact same subsequence can also be found when reading the reversed string from left to right, then you're essentially reading the original string from right to left and getting the same result. This only happens with palindromes!

## Concrete Example Walkthrough

Original: "racecar"  
Reversed: "racecar"

Any common subsequence, say "racr":

- In original: positions 0,1,2,6 → **racecar**
- In reversed: positions 0,1,2,6 → **racecar**

Since the string is the same as its reverse, any subsequence is automatically palindromic.

Or take: "abcde" vs "edcba"

- Only single characters can be common subsequences
- "a" appears in both → palindromic ✓

- "e" appears in both → palindromic ✓
- No longer common subsequences exist because no multi-character sequence can survive the complete reversal