Prompt Given a sequence of path fragments or patterns as passed to "-find", transform all path fragments into glob patterns. Pass through existing patterns untouched For example, given find paths of: ['foo.txt', 'pp:root/somedir']... transform that into: ['sh:**/*foo.txt*/**',

'pp:root/somedir'|

Check each path in the input sequence to determine if it contains any glob special characters ("", "!").

2. If a path contains any of these characters, add it to the result as-is. If a path does not contain glob characters, normalize it by stripping any trailing \(\gamma\). 5. Collect all transformed paths into a tuple and return it

CoT fails to capture prompt's pattern transformation rules. Generated Code

def make find paths(find paths):

transformed = [] for path in find paths: if any(c in path for c in {'*', '?', '['}):

transformed.append(path) alca.

stripped path = path.rstrip('/') transformed.append(f"{stripped path}/**")

return tuple(transformed)