Dihedral Group

- No matter how many Dihedral actions are applied, it will always boil down to a rotation, and possibly a reversal, of the sequence.
  - So just look for a subsequence, forward and backward

```
def is_subsequence( index:int, increment:int ) -> bool:
    """
    Check if the test sequence is a subsequence of the dihedral sequence
    index (int): Index of test[0] in the dihedral sequence
    increment (int): Either 1 or n-1 (effectively -1)
    """
    ok = True
    for x in test:
        if dihedral[index]==x:
            index += increment
            index %= n # Going off the end? Go back.
        else:
            ok = False
            break
    return ok
```

```
n, m = (int(x) for x in input().strip().split())
dihedral = [int(x) for x in input().strip().split()]
test = [int(x) for x in input().strip().split()]

# Locate test[0] in the dihedral sequence
start = None
for i in range(n):
    if dihedral[i]==test[0]:
        start = i
        break

if start is None: # test[0] not in the dihedral sequence? Fail.
    ok = False
else:
    ok = is_subsequence( start, 1 ) # Forwards
    if not ok:
        ok = is_subsequence( start, n-1 ) # Backwards

print( 1 if ok else 0 )
```