(Q14) Finding an optimal edit sequence to change UMSM to SUMOMISE by following the space optimal DP technique i.e., Half^{m} (i,j) function as discussed.

We will assume A[] = UMSM and use index i for this, and B[] for SUMOMISE, and use index j for this.

constructing Edit (4,8) for A[1...4] and B[1...8] using the recurrence relation as discussed:

	_	U	M	S	M
_	0	1	2	3	4
S	1	1	2	2	3
U	2	1	2	3	3
M	3	2	1	2	3
O	4	3	2	2	3
M	5	4	3	3	2
I	6	5	4	4	3
S	7	6	5	4	4
E	8	7	6	5	5

Table 1

From this table we find out that the smallest Edit Sequence from A → B is 5.

There exist two parts i.e., 2 Edit sequence from A → B, which we found out from the table-1. Both gives smallest Edit Sequence of length 5.

one of them being -

UMSM

↓ insert S

S UMSM

↓ no change

SU MSM

↓ no change

SUM SM

↓ replace S with O

SUMO[redacted]M

no change →

SUMOM

↑ insert I

SUMOMI

↑ insert S

SUMOMIS

↑ insert E

SUMOMISE

Now we will show this exact changes with the help of half( ) function call and show what length of B matches to some optimal Edit Sequence of A.

Defining $half^4(4,8)$ = length of the first part of B to which $A[1...4/2]$ is changed to some optimal Edit Sequence of $A[1...4] \rightarrow B[1...8]$

We will calculate the below table using the recurrence of half( ) as discused.

<u>$half^4(4,8)$ -</u>

	_	U	M	S	M
_	∞	∞	0	0	0
S	∞	∞	1	0	0
U	∞	∞	2	0	0
M	∞	∞	3	3	0
O	∞	∞	4	3	3
M	∞	∞	5	3	3
I	∞	∞	6	3	3
S	∞	∞	7	6	3
E	∞	∞	8	6	3

We got the value of $half^4(4,8) = 3$ which means $B[1...3]$ is the part to which $A[1..2]$ is changed to.

① UM $\longrightarrow$ SUM

② SM $\longrightarrow$ OMISE

Now we will calculate Edit (i,j) and $half^m(i,j)$ recursively on the above two sequences to get the entire mapping.

Solving for 1:

Edit (2,3)

	—	U	M
—	0	1	2
S	1	1	2
U	2	1	2
M	3	2	1

Edit (2,3) = length of the smallest edit sequence from UM → SUM is 1.

$half^2(2,3)$

	—	U	M
—	∞	0	0
S	∞	1	1
U	∞	2	2
M	∞	3	2

$half^2(2,3) = 2$, i.e., A[1] is changed to B[1...2]

U → SU

M → M { This is a no change operation, so we won't count it.

Edit() and half() for U → SU

Edit (1,2)

	—	U
—	0	1
S	1	1
U	2	1

Edit (1,2) = 1

length of smallest Edit Seq. for U → SU

$half^1(1,2)$

	—	U
—	0	0
S	1	0
U	2	1

$half^1(1,2) = 1$ i.e., A[0] maps to B[1]

— → S (operation 1: insert)

U → U (no change)

Solving for 2

Edit (2,5)

	–	S	M
–	0	1	2
O	1	1	2
M	2	2	1
I	3	3	2
S	4	3	3
E	5	4	4

Edit (2,5) = 4,
length of smallest Edit
Sequence from SM → OMISE

$half^2(2,5)$

	–	S	M
–	∞	0	0
O	∞	1	1
M	∞	2	1
I	∞	3	1
S	∞	4	1
E	∞	5	1

$half^2(2,5) = 1$ i.e.,
A[1] maps to B[1]
S → O (operation 2: replacement)
M → MISE

Edit() and half() for M → MISE

Edit (1,4)

	–	M
–	0	1
M	1	0
I	2	1
S	3	2
E	4	3

Edit(1,4) = 3,
length of the smallest edit sequence from
M → MISE

$half^1(1,4)$
here $m/2 = 0$, ie A[]
will not have any substring
to map to.
So from Edit_Distance (1,4)
we can find out the changes
which are as follows:

M
↓ no change
M
↓ insert I
MI
↓ insert S
MIS
↓ insert E
MISE

operation 3, 4, 5: Three insertions

Thus we got 5 operations of changing UMSM → SUMOMISE, which was also given by Table-1 and the mapping of each such change is also shown by the operation numbers.