Implement w/ 2xn array C:

\[ C[1][1..n] : \text{best time through } S_1_{1..n} \]
\[ C[2][1..n] : \text{'''''' through } S_2_{1..n} \]

Make_Car(a_1, a_2, t_1, t_2, n). // a_1, a_2: lists of station times
// t_1, t_2: '''''' transfer
// n: number of stations.

\[ C[1][0] = a_1 \] //time through \( S_1_0 \)
\[ C[2][0] = a_2 \] //'''''' \( S_2_0 \)

for (j = 2 to n)
//times through \( S_{1,j} \)

\[ \text{same}_1 = C[1][j-1] + a_{1,j} \] //coming from line 1
\[ \text{transfer}_1 = C[2][j-1] + t_{1,j-1} + a_{1,j} \] //transferring
//from line 2

if (\text{same}_1 \leq \text{transfer}_1)

\[ C[1][j] = \text{same}_1 \]
\[ \text{prevline}[1][j] = 1 \] //best prev station on line 1

else

\[ C[1][j] = \text{transfer}_1 \]
\[ \text{prevline}[1][j] = 2 \] //best prev station on line 2

//times through \( S_{2,j} \)

\[ \text{same}_2 = C[2][j-1] + a_{2,j} \] //coming from line 2
\[ \text{transfer}_2 = C[2][j-1] + t_{2,j-1} + a_{2,j} \] //transferring
//from line 1.
if (same2 < transfer2)

\[ c_{z2j_i} = \text{same2} \]
\[ \text{prevline} [2][j_i] = 2 \] // best prev station on line 2

else

\[ c_{z2j_i} = \text{transfer2} \]
\[ \text{prevline} [2][j_i] = 1 \] // best prev station on line 1

\%/end for

// Final min cost?

if (c_{E1}[n] < c_{E2}[n])

\[ c^* = c_{E1}[n] \]
\[ \text{lastline} = 1 \] // last station on line 1

else

\[ c^* = c_{E2}[n] \]
\[ \text{lastline} = 2 \] // last station on line 2
Try for example.

(From line 1, from line 2)

\[ C = \begin{array}{cccc}
1 & 7 & (16, 19) & (19, 17) & (21, 25) & (29, 32) \\
2 & 8 & (14, 13) & (25, 19) & (22, 23) & (29, 27) \\
\end{array} \]

Look at previous entries of table to fill next entry

\[ C^* = \min (29, 27) = 27 \]

Now, how to find the sequence of lines to visit?

Can see this sequence in the C array.

First, last line to visit?

= line that yields \( C^* \)

<add to code>

Now, sequence? Can see this in C array.
How to keep track of this while filling C?

Another array: prevline

prevline [j] (j = 2...n): previous best line before S_{1,j}
prevline [j] : "" : "" : "" : S_{2,j}

<add to code>

\[
\begin{array}{cccccc}
1 & 2 & 3 & 4 & 5 \\
-1 & 1 & 2 & 1 & 1 \\
-1 & 2 & 2 & 1 & 2 \\
\end{array}
\]

Now trace prevline starting from last line to -1.

last line = 2 \Rightarrow 2 \leftarrow 2 \leftarrow 1 \leftarrow 2 \leftarrow 2 \leftarrow -1

(station 5)

\[
\text{prevline [2][5]} = 2
\]
\[
\therefore \text{best line at station 4 was 2}
\]
Run Time: n Stations?

to fill C[J][J]: for loop from 1 to n : O(n)
" " prekline C[J][J] : O(n)

Total time: O(n)