To recursively characterize problem 1, let $V[i][0]$ be the maximum amount of treasure that can be collected on any route ending on the left side of hall segment i (including the treasure in that ending position), and similarly let $V[i][1]$ be the maximum amount of treasure that can be collected on any route ending on the right side of hall segment i. Also, let S be the array of left-side treasure, T be the array of right-side treasure, and G be the array of guardian fees.

Then

$$V[i][j] = \begin{cases} S[0] & \text{if } i = 0 \text{ and } j = 0 \\ T[0] & \text{if } i = 0 \text{ and } j = 1 \\ S[i] + \max(V[i-1][0], V[i-1][1] - G[i-1]) & \text{otherwise, if } j = 0 \\ T[i] + \max(V[i-1][0] - G[i-1], V[i-1][1]) & \text{otherwise} \end{cases}$$

The final answer would be the maximum of $V[n-1][0]$ and $V[n-1][1]$.

To recursively characterize the one-way GPS problem, let E be an $n-1 \times m$ array giving the travel times of the east-bound road segments. For example, $E[0][1]$ gives the travel time from intersection $(0,1)$ to intersection $(1,1)$. In the example given, $E[0][1] = 2$.

Similarly let N be a $n \times m-1$ array giving the travel times of the north-bound road segments. For example, $N[1][1]$ gives the travel time from intersection $(1,1)$ to intersection $(1,2)$. In the example given, $N[1][1] = 6$.

Let T be a $n \times m$ array for computing the best travel times to any intersection such that $T[i][j]$ is the least travel time from $(0,0)$ to (i,j). This problem can be recursively characterized by

$$T[i][j] = \begin{cases} 0 & \text{if } i = 0 \text{ and } j = 0 \\ T[i][j-1] + N[i][j-1] & \text{if } i = 0 \text{ and } j > 0 \\ T[i-1][j] + E[i-1][j] & \text{if } i > 0 \text{ and } j = 0 \\ \min(T[i][j-1] + N[i][j-1], \\ \quad T[i-1][j] + E[i-1][j]) & \text{otherwise} \end{cases}$$

The final answer is the value in $T[n-1][m-1]$.