- **Algorithm**

  First, we sort bodies and coffins by length so that $x_1 \leq x_2 \leq ... \leq x_n$ and $y_1 \leq y_2 \leq ... \leq y_m$ just as what we did in greedy algorithm.

  Let $f(i,j)$ denote the minimum badness of fit for the first $i$ bodies and the first $j$ coffins, $1 \leq i \leq m$ and $1 \leq j \leq n$. The base cases are when $i = j$, simply assigning $x_i$ to $y_i$. Then, for each $i < j$, we calculate $f(i,j)$ from $f(i,i+1)$ to $f(i,n)$.

  For calculating $f(i,j)$ we have two cases,

  - **Case 1: assign $x_i$ to $y_j$.** Then we are left with a subproblem of assigning $(i-1)$ bodies to $(j-1)$ coffins, whose badness of fit is denoted by $f(i-1,j-1)$. Together with our newest match, we get $f(i-1,j-1)+ \mid x_i - yj \mid$ as badness of fit.
  - **Case 2: not assign $x_i$ to $y_j$.** Then an optimal solution must assign $x_i$ to some previous coffin, which is included in a subproblem with $f(i,j-1)$ as badness of fit.

  As what is analyzed above, we drive the equation for calculating $f(i,j)$ for any $i < j$

  $$f(i,j) = min(f(i-1,j-1)+ \mid x_i - y_j \mid, f(i,j-1)).$$

  The above algorithm can be illustrated as follows.

| m $\mid n$ | 1 | 2 | 3 | 4 | ... |
|---|---|---|---|---|---|
| 1 | $\mid x_1 - y_1 \mid$ | NA | NA | NA | ... |
| 2 | $\mid x_1 - y_2 \mid$ | $f(1,1)+ \mid x_2 - y_2 \mid$ | NA | NA | ... |
| 3 | $\mid x_1 - y_3 \mid$ | $f(1,2)+ \mid x_2 - y_3 \mid$ or $f(2,2)$ | | NA | ... |
| 4 | $\mid x_1 - y_4 \mid$ | $f(1,3)+ \mid x_2 - y_4 \mid$ or $f(2,3)$ | | | ... |
| ... | ... | ... | | | ... |
| n | $\mid x_1 - y_n \mid$ | $f(1,n-1)+ \mid x_2 - y_n \mid$ or $f(2,n-1)$ | | | ... |

  And our ultimate solution lays at $f(m,n)$, which is at the right bottom corner of the table.

- **Correctness**

  The above DP algorithm is optimal as it has the following three traits.

  - **Complete Choice**: Suppose there exists an optimal solution. According to what has been proved in greedy case of Homework 1 Question 3, badness of fit is minimized if and only if $(c-a)(d-b) \geq 0$ for every $x_a$ assigned to $y_b$ and $x_c$ assigned to $y_d$. For each $x_i$, either assigning it to $y_j$ is optimal or not assigning it to $y_j$ is optimal. Thus, we have considered all possibilities that might contain optimal solution.
  - **Inductive Structure**: We have two cases, each of which contains a subproblem. Because we are sure that bodies and coffins should be matched in an order, there is no external constraints how we solve each subproblem.
  - **Optimal Substructure**: The total badness of fit is the sum of every matching pair in solution P. If we solve a subproblem optimally, then combining it with the same new matching pair will generate an optimal solution to P' according to standard contradiction argument.

- **Complexity**

  The overall time complexity of the algorithm is $\mathbf{O}(mn + mlogm)$.

  To sort the bodies and coffins requires O($mlogm$) and O($nlogn$), respectively. To calculate for the $m$ by $n$ array requires O($mn$) as comparing two cases for each cell of the array requires O(1). Because we have no evidence to show that O($mn$) is greater than O($nlogn$) or not, our final complexity will be O($mn + mlogm$).