

Dinamikus programozás

Mi az algoritmus célja?:

A dinamikus programozást olyan helyzetekben használjuk, amikor döntések sorozatát kell meghoznunk az optimális megoldás eléréséhez. Gyakran előfordul, hogy egy döntés meghozatala után a fennmaradó részprobléma hasonló lesz az eredetihez. Ez a módszer különösen hatékony, ha ugyanaz a részprobléma többféle döntési sorozat eredményeként is előállhat.

Az algoritmus lépései:

Hozzunk létre egy dp mátrixot, ahol $dp[i][j]$ a két sztring első i és j karaktere közötti leghosszabb közös részsorozat hosszát tárolja. Ha az aktuális karakterek megegyeznek, $dp[i][j] = dp[i-1][j-1] + 1$. Ha nem egyeznek, $dp[i][j] = \max(dp[i-1][j], dp[i][j-1])$. A mátrix utolsó cellája tartalmazza a teljes leghosszabb közös részsorozat hosszát.

Az algoritmus magyarázata:

dp mátrix:

- A mátrix sorai az első sztring karaktereit, az oszlopok a második sztring karaktereit reprezentálják.
- A $dp[i][j]$ tárolja a leghosszabb közös részsorozat hosszát az i -edik és j -edik pozícióig.

Algoritmus lépései:

- A mátrixot nullákkal inicializáljuk.
- Ha két karakter egyezik, a bal-felső szomszéd értékéhez 1-et adunk.
- Ha nem egyeznek, a bal vagy a felső szomszéd közül a nagyobbbat vesszük.

Bemenet:

- Két sztringet kér be a felhasználótól.

Kimenet:

- A leghosszabb közös részsorozat hossza
-