LIS:

Sea:

$A = [2, 5, 3, 8]$ // 1-indexed

$n = len(A)$

[2] 1

[2 5] [5] 2

[2 5 3] [2 3] 2

[2 8] [2 5 8] [2 3 8] 3

Lis = MAX(1,2,2,3) = 3

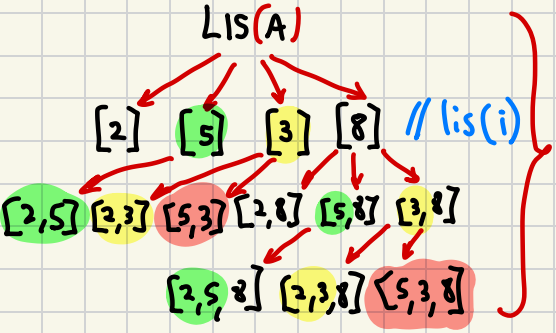
$$lis(i) = \begin{cases} 0, & i = 0 \\ 1 + \max\{lis(j) : A[j] < A[i],\ j = 0\} \end{cases}$$

// Lista vacía

// Lista crecientes terminando en i

$$LIS(A) = \max\{lis(i) : i \in [1, n]\}$$

// Lista crecientes terminando en cualquier i



Se pueden utilizar lis(i) ya calculados (eg y)

Sea:

list dp[n]

función lis base

Top-Down — memoización

```
def lis(i):
  if dp[i]!=0
    return dp[i]
  else:
    for j=[1...i-1]:
      if A[j]<A[i]:
        dp[i]=max(dp[i],1+dp[j])
dp=zeros(n)
return max{lis(i)}
```

Bottom-up — tabulación

```
for i=[1...n]:
  dp[i]=1
  for j=[1...i-1]:
    if A[j]<A[i]:
      dp[i]=max(dp[i],1+dp[j])
return max{dp[i]}
```