

# Análisis de algoritmos 2022-1

## Tarea 8.

Fecha de entrega: 17 de diciembre de 2021.

### 1. Tarea

1. 10pt. Dada una secuencia de números  $S = s_1, s_2, \dots, s_n$ , una *subsecuencia creciente* es una subsecuencia de la original, con la propiedad de que sus elementos aumentan monótonamente de forma estricta en valor (es decir, todo elemento es mayor que los previos a él). P. Ej. Si  $S = 4, 2, -1, 3, 2, 3$ , algunas de sus subsecuencias crecientes son: la secuencia 4; la secuencia 2, 3 y la secuencia  $-1, 2, 3$ .

Considera el problema de, dada una secuencia de entrada  $S = s_1, s_2, \dots, s_n$ , calcular alguna subsecuencia creciente de longitud máxima posible. En el ejemplo anterior, la subsecuencia  $-1, 2, 3$  sería una respuesta válida (para ese ejemplo la subsecuencia de longitud máxima es única, pero en general no tiene porqué serlo).

Tip: considera la familia de subproblemas  $\{\mathcal{I}_i(1 \leq i \leq n)\}$  tal que  $\mathcal{I}_i$  es el problema de determinar la longitud de la subsecuencia creciente más larga de  $S$ , tal que el  $s_i$  es el último elemento de esa secuencia.

- a) 3pt. Propón la ecuación de Bellman del problema, demostrando que es correcta por inducción.
- b) 3pt. Propón la versión recursiva con memorización del algoritmo que resulta de aplicar la ecuación de Bellman. Analiza su corrección y complejidad.
- c) 3pt. Propón la versión iterativa de programación dinámica del algoritmo anterior. Analiza su corrección y complejidad.
- d) 1pt. Propón el algoritmo que, a partir de la(s) tabla(s) generadas por el algoritmo anterior, calcula una subsecuencia creciente de longitud máxima (los algoritmos anteriores calculan sólo su longitud).