



Nombre y apellidos:

DNI:

- Duración del examen: 1 hora y 45 minutos.
- Todas las hojas entregadas deben tener nombre y DNI del alumno y Nº de página/Total de páginas.
- Las respuestas a los ejercicios deberán escribirse en pseudocódigo y estar claramente justificadas, es decir, acompañadas de una breve y clara explicación.
- En todas las preguntas pueden incluirse operaciones auxiliares, debidamente justificadas, si se considera necesario.
- En el apartado a) de todas las preguntas deben explicarse claramente las operaciones básicas del TAD (tipos de entrada y salida de las mismas y ecuaciones de definitud) y describir su funcionamiento.

(4 puntos) Ejercicio 1.-

a) Dar la especificación del TAD PILA[ELEMENTO].

(0,25 pts)

Suponiendo conocidas las operaciones $=$: elemento elemento \rightarrow bool, que comprueba si dos elementos son iguales, escribir en pseudocódigo las siguientes operaciones (pueden ser parciales) partiendo únicamente de la especificación del TAD PILA[ELEMENTO]:

- a) *crea_copia*: pila \rightarrow pila, crea una copia de la pila dada. Es decir, con los mismos elementos y en el mismo orden. (0,5 pts)
- b) *parte_arriba*: pila pila \rightarrow bool, comprueba si los elementos de la primera pila están en la parte superior de la segunda y en el mismo orden. (0,75 pts)
- c) *es_trozo*: pila pila \rightarrow bool, comprueba si la primera pila es un trozo de la segunda. Es decir, todos los elementos de la primera están en la segunda en el mismo orden y sin otros elementos entre ellos. (1,25 puntos)

- d) Escribir **el apartado b)** utilizando un algoritmo **recursivo** (o iterativo si se ha implementado recursivo). (1,25 punto)



(4 puntos) Ejercicio 2.-

- a) Dar la especificación del TAD COLA[ELEMENTO]. (0,25 pts)
b) Dar la especificación del TAD LISTA2[ELEMENTO]. (0,25 pts)

Escribir en pseudocódigo las siguientes operaciones (pueden ser parciales), partiendo únicamente de laS especificaciones de los tipos abstractos de datos COLA[ELEMENTO] y LISTA[ELEMENTO]:

- a) *crea_copia*: cola → cola, crea una copia de la cola dada. Es decir, con los mismos elementos y en el mismo orden. (0,5 pts)
b) *parte_de*: cola lista → boolean, comprueba si la cola es un trozo de la lista. Es decir, todos los elementos de la cola están en la lista en el mismo orden y sin otros elementos entre ellos. (1,25 pts)
c) *primero_en_cola*: lista cola → lista, obtiene una lista con los elementos de la lista dada pero situando los que pertenecen a la cola al inicio no necesariamente en el mismo orden. (1,25 pts)

ejemplo: cola → E1 E3 E5 E8 E0

lista → E6 E5 E1 E9 E0 E3 E4 E3

lista *primero_en_cola* → E3 E3 E0 E1 E5 E6 E9 E4

- a) Escribir el **apartado c)** utilizando un algoritmo **recursivo** (o iterativo si se ha implementado recursivo). (1 pto)
b) Escribir el **apartado a)** utilizando la implementación con **memoria dinámica** del TAD COLA vista en clase. (1,5 pts)