

Estructura de Datos y Algoritmos
Grado Ingeniería Informática
Doble Grado Ingeniería Informática
y Administración de Empresas
Universidad Carlos III de Madrid
CURSO 2019-2020



Convocatoria Extraordinaria. 26 Junio 2020

Instrucciones:

- Duración: **40 minutos** (para desarrollar y subir tu solución).
- Lee atentamente el problema. Vuelve a leer el problema una segunda vez.
- Si tienes alguna duda pregunta en el chat. **NO PREGUNTES COSAS QUE ESTÁN EN EL ENUNCIADO. Sólo se responderán preguntas los 10 primeros minutos.**
- Descarga el fichero **problema1y2.py** y completa tu solución. Pon atención en qué carpeta vas a descargar y modificar el fichero. **NO RENOMBRES EL FICHERO.**
- Este fichero contiene una clase Test (unittest) que te permitirá comprobar si tu solución es correcta o no. **NO BORRES NI MODIFIQUES Test.**
- Cuando queden 5 minutos el profesor, te avisará para que subas tu solución (fichero **problema1y2.py**) a la actividad **Entrega Problemas 1 y 2**. Las soluciones enviadas fuera de tiempo no serán evaluadas.
- Es tu responsabilidad comprobar que has subido la versión correcta del fichero.

Problema 1 (15 puntos):

Sea DList la implementación de lista doblemente enlazada (una versión simplificada con los métodos necesarios para inicializar una lista con elementos). Completa su función **skipMremoveN**, que recibe dos enteros M y N positivos. La función debe recorrer la lista, conservando M nodos, y entonces borrando los N nodos siguientes. El proceso debe continuar hasta que finalice el recorrido. La función debe modificar la lista. La función no devuelve nada.

Para desarrollar la solución no puedes usar listas auxiliares, de ningún tipo. Solo puedes manipular y modificar la lista invocante.

Pon atención a los siguientes ejemplos:

- Input: M = 1, N = 1, lista:

1<->2<->3<->4<->5<->6<->7<->8<->9<->10<->11<->12<->13<->14

- Output: lista: 1<->3<->5<->7<->9<->11<->13

- Input: M = 1, N = 2, lista:

1<->2<->3<->4<->5<->6<->7<->8<->9<->10<->11<->12<->13<->14

- Output: lista: 1<->4<->7<->10<->13

- Input: M = 1, N = 3, lista:

1<->2<->3<->4<->5<->6<->7<->8<->9<->10<->11<->12<->13<->14

- Output: lista: 1<->5<->9<->13

- Input: M = 1, N = 4, lista:

1<->2<->3<->4<->5<->6<->7<->8<->9<->10<->11<->12<->13<->14

- Output: lista: 1<->6<->11

- Input: M = 2, N = 2, lista:

1<->2<->3<->4<->5<->6<->7<->8<->9<->10<->11<->12<->13<->14

- Output: lista: 1<->2<->5<->6<->9<->10<->13<->14

- Input: M = 2, N = 3, lista:

1<->2<->3<->4<->5<->6<->7<->8<->9<->10<->11<->12<->13<->14

- Output: lista: 1<->2<->6<->7<->11<->12

- Input: M = 2, N = 4, lista:

1<->2<->3<->4<->5<->6<->7<->8<->9<->10<->11<->12<->13<->14

- Output: lista: 1<->2<->7<->8<->13<->14

- Input: M = 3, N = 2, lista:

1<->2<->3<->4<->5<->6<->7<->8<->9<->10<->11<->12<->13<->14

- Output: lista: 1<->2<->3<->6<->7<->8<->11<->12<->13

- Input: M = 3, N = 3, lista:

1<->2<->3<->4<->5<->6<->7<->8<->9<->10<->11<->12<->13<->14

- Output: lista: 1<->2<->3<->7<->8<->9<->13<->14

- Input: M = 3, N = 4, lista:

1<->2<->3<->4<->5<->6<->7<->8<->9<->10<->11<->12<->13<->14

- Output: lista: 1<->2<->3<->8<->9<->10

- Input: M = 3, N = 11, lista:

1<->2<->3<->4<->5<->6<->7<->8<->9<->10<->11<->12<->13<->14

- Output: lista: 1<->2<->3

- Input: M = 14, N = 1, lista:

1<->2<->3<->4<->5<->6<->7<->8<->9<->10<->11<->12<->13<->14

- Output: lista: 1<->2<->3<->4<->5<->6<->7<->8<->9<->10<->11<->12<->13<->14

Problema 2 (5 puntos): En la implementación de tu función, añade comentarios que respondan a los siguientes preguntas:

- ¿Cuál es la complejidad del método?. Razona (1 punto)
- Indica cuál es el mejor caso y su complejidad. Explica por qué (2 puntos).
- Indica cuál es el peor caso y su complejidad. Explica por qué (2 puntos).