



Universidad Nacional
de San Martín

Licenciatura en Ciencia de Datos

Algoritmos II

Parte 1

RECURSIÓN

Ejercicios

Ejercicio 16

Definir la función sublista, que dada una lista de enteros, un número que represente una posición y otro número que represente una longitud, devuelva una lista de enteros (que se basa en la lista dada) que comience en la posición dada y que tenga la longitud dada desde esa posición (pila)

Ejercicio 16 bis

Definir la función sublista, que dada una lista de enteros, un número que represente una posición y otro número que represente una longitud, devuelva una lista de enteros (que se basa en la lista dada) que comience en la posición dada y que tenga la longitud dada desde esa posición (cola)

Ejercicio 16 bis bis

Definir la función sublista, que dada una lista de enteros, un número que represente una posición y otro número que represente una longitud, devuelva una lista de enteros (que se basa en la lista dada) que comience en la posición dada y que tenga la longitud dada desde esa posición (iterativa)

Ejercicio 17

Definir la función esPalindromo, que dada una lista de enteros, retorne si es o no es palíndromo, utilizando recursividad explícita.

Ejercicio 18

Definir la función aplanar, que dada una lista de listas de enteros, retorne un lista de enteros que corresponda a la concatenación de elementos-lista de la lista original.

Por ejemplo: `aplanar([[5,7], [], [3,7,2], [9]]) = [5,7,3,7,2,9]`.

Ejercicio 19

Definir la función quicksort, que dada una lista de enteros, retorne la lista original, pero con sus elementos ordenados, utilizando el método de ordenamiento quicksort

Ejercicio 20

Implementar la función recursiva posiciones_pares, que dado una lista de enteros, imprima el contenido de sus posiciones pares

Ejercicio 21

Implementar la función recursiva producto_escalar, que dados dos vectores de enteros, retorne un entero que represente el producto escalar de ambos (pila)

Ejercicio 21 bis

Implementar la función recursiva producto_escalar, que dados dos vectores de enteros, retorne un entero que represente el producto escalar de ambos (cola)

Ejercicio 21 bis bis

Implementar la función recursiva producto_escalar, que dados dos vectores de enteros, retorne un entero que represente el producto escalar de ambos (iterativo)

Ejercicio 22

Implementar la función recursiva busquedaBinaria, que dado un vector ordenado de enteros y un entero dado, retorne si el entero dado pertenece a alguna posición del vector. La búsqueda deberá efectuarse con la técnica de búsqueda binaria.

Ejercicio 22 bis

Implementar la función iterativa busquedaBinaria, que dado un vector ordenado de enteros y un entero dado, retorne si el entero dado pertenece a alguna posición del vector. La búsqueda deberá efectuarse con la técnica de búsqueda binaria.

Ejercicio 23

Definir dos procedimientos `contar_hacia_atras_par` y `contar_hacia_atras_impar` con recursión mutua que dado un número `n`, imprima por pantalla todos los números de manera descendente hasta llegar a uno.

Por ejemplo:

```
contar_hacia_atras_par(5)
```

Salida:

5

4

3

2

1

Ejercicio 24

Definir la función partes, que dada una lista de enteros, retorne una lista de listas de enteros, en que cada elemento de la lista resultado sea cada una de las sublistas de la lista original (respetando la posición).

**Por ejemplo: partes([6,2,3]) =
[[],[6],[2],[3],[6,2],[6,3],[2,3],[6,2,3]].**

(no necesariamente en ese orden)

Ejercicio 25

Definir la función todosConTodos, que dadas dos listas de enteros, retorne una lista de pares de enteros que formen parte de combinaciones de elementos de la primera lista con elementos de la segunda lista

Ejercicio 26

Definir una versión recursiva a la función penultimo, que dada una listas de enteros, retorne su penúltimo elemento.

Ejercicio 27

Definir la función primeros, que dada una lista de enteros y un número n dado, retorne los primeros n elementos de la lista dada.