

Práctico 9 – Recursión, Archivos de Texto, Variado

Ejercicio 1

Realizar un programita que lea un archivo de texto llamado datos.txt ya existente, y muestre en pantalla todas las líneas de texto que contiene.

Ejercicio 2

Realizar un programita que cargue un vector con 50 números obtenidos de un archivo de texto llamado nros.txt y los muestre en pantalla. En el archivo los números los deberá poner uno por línea a mano.

Ejercicio 3

Realizar un programita que pida 5 palabras al usuario, y las guarde en un archivo de texto que exista, llamado palabras.txt

Ejercicio 4

Realizar un programita que guarde en un archivo de texto, y que lo cree si no existe, 50 números al azar entre 0 y 1. Uno por línea.

Ejercicio 5

Realizar una función que tome como argumento un número entero n , y devuelva la suma de todos los números naturales hasta ese n . Utilice recursión para resolverlo.

Ejercicio 6

Realizar una función llamada Fibonacci, que implemente la misma, utilizando recursión. Recordar la función de Fibonacci

$$\begin{aligned}f_0 &= 0 \\f_1 &= 1 \\f_n &= f_{n-1} + f_{n-2}\end{aligned}$$

Ejercicio 7

Implementar una función llamada Longitud, que tome como parámetro un vector de números enteros y determine la cantidad de elementos (*como un lenght*) pero utilizando recursión para resolverla.

Ejercicio 8

Realizar una función que tome como argumentos un vector y un número, y determine la cantidad total de veces que ese número se encuentra en el vector. Utilice recursión para resolverlo.

Ejercicio 9

Realizar un programa, que, habiendo implementado una clase Empleado, sencilla, con nombre y apellido, sueldo actual y antigüedad como atributos, y un método toString() además del constructor, lo utilice en el programa para crear un ArrayList de empleados. Pedir al usuario datos de empleados para ir cargando el ArrayList, hasta que ingrese una palabra “salir” para dejar de cargar, y entonces el programa mostrará todos los empleados que fueron cargados, uno por línea, todos sus datos.

Ejercicio 10

Supongamos que se tiene un vector de números enteros, y otro vector de Strings. Implementa un único método mostrarArray que pueda tomar como argumento cualquiera de esos dos vectores, e imprimirlo en pantalla. Ese mismo método podrá tomar cualquiera de los dos vectores, a pesar de que son de tipos distintos. Resolver utilizando la creación de un método genérico.

Ejercicio 11

Crear una clase muy simple llamada Animal, que tenga un método caminar, que imprima en pantalla "Caminando". Ahora, crear otra clase, descendiente de la misma, llamada Pajaro e implementar un solo método llamado volando, que imprima en pantalla "Volando".

Ejercicio 12

Implementar una clase Vehiculo que tendrá métodos como avanzar, detener y atributos como cantidad de ruedas, velocidad máxima, cantidad de asientos, modelo, marca. E implementar las siguientes clases descendientes: auto, camión y colectivo. Agregar los atributos que consideren específicos de cada uno.

Ejercicio 13

Suponiendo que en vacaciones, de los vehículos que se censan en un control policial, se observa un 20% de camiones, un 15% de colectivos y el resto de autos. Realizando hipótesis necesarias y adecuadas, estime la cantidad de turistas que ingresan por día. Proponga que otras cosas deberían censarse en los vehículos que ingresan. Sabiendo además, que en un día promedio entran entre 1.500 y 2.000 vehiculos. Necesitará utilizar random, vectores dinámicos.