

VT2 - PD2

Ejercicio 1 / calcular Factorial

Función factorial (n: entero): entero

Comienzo

Si $n = 0$ o $n = 1$ entonces // caso Base $0 = 1$

Devolver 1

Fin si

Devolver $n \cdot \text{factorial}(n-1)$ // Paso recursivo $n = f(n-1)$

Fin

2) Analizar orden del tiempo de ejecución

El tiempo de ejecución de algoritmos, es en orden $O(n)$ ya que es el peor caso. Esto se repite las veces que sea necesario dependiendo del valor de n , haciendo un llamado recursivo hasta llegar al caso base, cuando n es 0 o 1.

Ejercicio 2 / suma Lineal

Función sumaLineal (A: arreglo de enteros, n: Entero) Entero

Comienzo

Si $n = 1$ entonces // Caso Base, si n es igual a 1

Devolver $A[0]$

Fin si

// Suma el último elemento en la sublista de los primeros n elementos con

Devolver $\text{sumaLineal}(A, n-1) + A[n-1]$

Fin // La suma de los primeros $n-1$ elementos

2) Analisis orden del tiempo de ejecución

El tiempo de ejecución es en orden $O(n)$, esto debido a que es el peor caso. Llama de forma recursiva hasta llegar al caso base.

Ejercicio 3 Potencia de un número

Función potencia (base: Real, exponente: Entero): Real

Comienzo // caso Base: cualquier número elevado a la potencia de
si exponente = 0 entonces // 0 es 1
Devolver 1

Fin si

// caso base para exponentes negativos: invierte la base y cambia el signo
si exponente < 0 entonces del exponente

(E-1) // Devolver 1/potencia (base, -exponente)

Fin si

// Paso recursivo: multiplicar la base por la potencia de la base al
Devolver Base * potencia (base, exponente - 1)

Fin // exponente menos uno.

2) Análisis de Orden de ejecución tiempo de ejecución

El orden de ejecución del algoritmo es en $O(n)$, ya que es el peor caso. De modo que el algoritmo se ejecuta n veces, dependiendo del valor de n , hasta llegar al caso base.

Ejercicio 4 Invertir Array

Algoritmo Invertir (A, i, J):

Entradas: Un array A de enteros e Índices enteros no negativos i y J

Salida: Los elementos de A entre los índices i y J Invertidos

Comienzo // caso Base: cuando el índice i no es menor que J.

si $i < J$ entonces

Intercambiar $A[i]$ y $A[J]$

Invertir Array (A, i + 1, J - 1)

Fin si

Fin