

Practica 2 - Recursividad

- 1.- Calcule el cociente y el residuo que resultan de dividir dos números enteros.
- 2.- Calcule la suma de los dígitos de un número.

3.- La relación recurrente

$$x^n = \begin{cases} 1 & \text{si } n = 0 \\ x * x^{n-1} & \text{si } n > 0 \\ x^{n+1} / x & \text{si } n < 0 \end{cases}$$

define x como la potencia n -ésima de todos los enteros. Escribir un procedimiento recursivo llamado *Potencia* que calcule x a la n -ésima potencia para un entero x .

- 4.- El máximo común divisor de dos enteros positivos M y N se puede definir como:

$$MCD(M, N) = \begin{cases} N & \text{si } N \leq M \text{ y } M \bmod N = 0 \\ MCD(N, M) & \text{si } M < N \\ MCD(N, M \bmod N) & \text{en otro caso} \end{cases}$$

Escribir una función recursiva que permita calcular el máximo común divisor de dos enteros positivos.

- 7.- En general, el N -ésimo término de la sucesión de Fibonacci, se define como:

$$F(A, B, N) = \begin{cases} A & \text{si } N = 1 \\ B & \text{si } N = 2 \\ F(B, A + B, N - 1) & \text{si } N > 2 \end{cases}$$

Donde A y B son los números que originan la secuencia. Escribir un programa que compute el N -ésimo término de la sucesión de Fibonacci usando diferentes números para originar la secuencia.

- 8.- La función Q se define como:

$$Q(N) = \begin{cases} 1 & \text{si } N \leq 2 \\ Q(N - Q(N - 1)) & \text{si } N > 2 \end{cases}$$

Escribir dos procedimientos, uno recursivo y otro no recursivo, para computar el valor de esta función para varios valores de N . ¿Puedes escribir una función que haga uso de más memoria para así aumentar la velocidad de los cálculos recursivos?

9.- El valor del N-ésimo término del polinomio de Legendre se puede calcular usando las siguientes fórmulas:

$$P_0(x) = 1$$

$$P_1(x) = x$$

$$P_N(x) = \frac{2 * N - 1}{N} * x * P_{N-1}(x) - \frac{N - 1}{N} * P_{N-2}(x)$$

Escribir una función recursiva que compute el N-ésimo término del polinomio de Legendre.

10.- Escribir una función recursiva para computar la función de Ackermann, la cual se define como:

$$A(M, N) = \begin{cases} N + 1 & \text{si } M = 0 \\ A(M - 1, N) & \text{si } M \neq 0 \text{ and } N = 0 \\ A(M - 1, A(M, N - 1)) & \text{si } M \neq 0 \text{ and } N \neq 0 \end{cases}$$

donde M y N son números cardinales.

11.- Para cualesquiera enteros no negativos N y K, donde $0 \leq K \leq N$, el coeficiente binomial $C(N, K)$ se define como:

$$C(N, K) = \frac{N!}{K! * (N - K)!}$$

Esta fórmula verifica que:

1. $C(N, K) = C(N, N - K)$; es decir, la lista de coeficientes binomiales es simétrica con respecto a su valor central.
2. $C(N, N) = 1$ y por simetría $C(N, 0) = 1$.
3. $C(N, N - 1) = N$ y por simetría $C(N, 1) = N$.

El problema para computar $C(N, K)$ haciendo uso de la fórmula anterior estriba en que $N!$ crece muy rápidamente. Por lo que necesitamos un método alternativo para calcular dichos coeficientes. Para ello podemos hacer uso del hecho de que para cualquier K que satisfaga $1 \leq K \leq N - 1$:

$$C(N, K) = \frac{C(N, K - 1) * (N - K + 1)}{K}$$

Escribir un programa que calcule el coeficiente binomial $C(N, K)$ para varios N y K.

13.- En un cuadrado cuyo lado es a, se unen los puntos medios de sus 4 lados, formandose otro cuadrado cuyos puntos medios se unen también formando otro cuadrado, y así sucesivamente. Calcular la suma de los perímetros de los n primeros cuadrados así formados.

14.- Calcular el producto de los dígitos de un número

15.- Verificar si todos los dígitos de un número son iguales

16.- Verificar si los dígitos de un número están en escalera ascendente

17.- Convertir un número de base 10 a base b

18.- Calcule el m.c.m. de dos números