

# MODULARIDAD EN C++

- FUNCIONES
- PROCEDIMIENTOS
- RECURSIVIDAD



## EJERCICIOS

Estructura de Datos

# EJERCICIOS DE APRENDIZAJE

Para cada ejercicio propuesto, primero lleva a cabo un análisis detallado del problema con el fin de determinar qué información será requerida como entrada y qué información será generada como salida.

## Subprogramas: Funciones

1. Realizar una función que calcule la suma de dos números. En el algoritmo principal le pediremos al usuario los dos números para pasárselos a la función. Después la función calculará la suma y lo devolverá para imprimirlo en el algoritmo.
2. Realizar una función que valide si un número es impar o no. Si es impar la función debe devolver un verdadero, si no es impar debe devolver falso. Nota: la función no debe tener mensajes que digan si es par o no, eso debe pasar en el Algoritmo
3. Crea una función EsMultiplo que reciba los dos números pasados por el usuario, validando que el primer número múltiplo del segundo y devuelva verdadero si el primer número es múltiplo del segundo, sino es múltiplo que devuelva falso.
4. Realizar un programa que pida al usuario una frase y una letra a buscar en esa frase. La función debe devolver la cantidad de veces que encontró la letra. Nota: recordar el uso de la función Subcadena().
5. Realizar una función que reciba un número ingresado por el usuario y averigüe si el número es primo o no. Un número es primo cuando es divisible sólo por 1 y por sí mismo, por ejemplo: 2, 3, 5, 7, 11, 13, 17, etc. Nota: recordar el uso del MOD.

## Subprogramas: Procedimientos

1. Realizar un procedimiento que permita intercambiar el valor de dos variables de tipo entero. La variable A, debe terminar con el valor de la variable B.
2. Crear un procedimiento que calcule la temperatura media de un día a partir de la temperatura máxima y mínima. Crear un programa principal, que utilizando un procedimiento, vaya pidiendo la temperatura máxima y mínima de n días y vaya mostrando la media de cada día. El programa pedirá el número de días que se van a introducir.

3. Realizar un procedimiento que permita realizar la división entre dos números y muestre el cociente y el resto utilizando el método de restas sucesivas.

El método de división por restas sucesivas consiste en restar el divisor con el dividendo hasta obtener un resultado menor que el divisor, este resultado es el residuo, y el número de restas realizadas es el cociente. Por ejemplo:  $50 / 13$ :

$50 - 13 = 37$  una resta realizada

$37 - 13 = 24$  dos restas realizadas

$24 - 13 = 11$  tres restas realizadas

dado que 11 es menor que 13, entonces: el residuo es 11 y el cociente es 3.

4. Escribir un programa que procese una secuencia de caracteres ingresada por teclado y terminada en punto, y luego codifique la palabra o frase ingresada de la siguiente manera: cada vocal se reemplaza por el carácter que se indica en la tabla y el resto de los caracteres (incluyendo a las vocales acentuadas) se mantienen sin cambios.

a	e	i	o	u
@	#	\$	%	*

Realice un subprograma que reciba una secuencia de caracteres y retorne la codificación correspondiente. Utilice la estructura “según” para la transformación.

Por ejemplo, si el usuario ingresa: Ayer, lunes, salimos a las once y 10.

La salida del programa debería ser: @y#r, l\*n#s, s@l\$m%\$ @ l@s %nc# y 10.

NOTA: investigue el uso de la función concatenar.

## Recursión

1. Escribir una función recursiva que devuelva la suma de los primeros N enteros.

## EJERCICIOS DE APRENDIZAJE EXTRA

### Funciones:

1. Realizar una función que calcule y retorne la suma de todos los divisores del número n distintos de n. El valor de n debe ser ingresado por el usuario.
2. Diseñar una función que reciba un número en forma de cadena y lo devuelva como número entero. El programa podrá recibir números de hasta 3 dígitos.  
Nota: no poner números con decimales

3. Crear una función llamada "Login", que recibe un nombre de usuario y una contraseña y que devuelve Verdadero si el nombre de usuario es "usuario1" y si la contraseña es "asdasd". Además, la función calcula el número de intentos que se ha usado para loguearse, tenemos solo 3 intentos, si nos quedamos sin intentos la función devolverá Falso.
4. Los empleados de una fábrica trabajan en dos turnos: Diurno y Nocturno. Se desea calcular el jornal diario de acuerdo con las siguientes reglas:
  - a) La tarifa de las horas diurnas es de \$ 90
  - b) La tarifa de las horas nocturnas es de \$ 125
  - c) En caso de ser feriado, la tarifa se incrementa en un 10% si el turno es diurno y en un 15% si el turno es nocturno.

El programa debe solicitar la siguiente información al usuario: el nombre del trabajador, el día de la semana, el turno (diurno o nocturno) y la cantidad de horas trabajadas. Además, debemos preguntarle al usuario si el día de la semana (lunes, martes, miércoles, etc.) era festivo o no, para poder calcular el jornal diario. Utilice una función para realizar el cálculo.

5. Realizar una función que calcule la suma de los dígitos de un número.

**Ejemplo:**  $25 = 2 + 5 = 7$

Nota: Para obtener el último número de un dígito de 2 cifras o más debemos pensar en el resto de una división entre 10. Recordar el uso de la función Mod y Trunc.

6. Realizar una función que reciba un número ingresado por el usuario y averigüe si el número tiene todos sus dígitos impares (ejemplo: 333, 55, etc.). Para esto vamos a tener que separar el número en partes (si es un número de más de un dígito) y ver si cada número es par o impar. Nota: recordar el uso de la función Mod y Trunc(). No podemos pasar el número a cadena para realizar el ejercicio.

7. Realizar una función que permita obtener el término n de la sucesión de Fibonacci.

La sucesión de Fibonacci es la sucesión de los siguientes números:

1, 1, 2, 3, 5, 8, 13, 21, 34, ...

Donde cada uno de los números se calcula sumando los dos anteriores a él. Por ejemplo:

La sucesión del número 2 se calcula sumando (1+1)

Análogamente, la sucesión del número 3 es (1+2),

Y la del 5 es (2+3),

Y así sucesivamente...

La sucesión de Fibonacci se puede formalizar de acuerdo a la siguiente fórmula:

Fibonacci (n) = Fibonacci (n-1) + Fibonacci (n-2) para todo  $n > 1$

Fibonacci (n) = 1 para todo  $n \leq 1$

Por lo tanto, si queremos calcular el término “n” debemos escribir una función que reciba como argumento el valor de “n” y que calcule la serie hasta llegar a ese valor.

Para conocer más acerca de la serie de Fibonacci consultar el siguiente link:

<https://quantdare.com/numeros-de-fibonacci/>

8. Realizar una función que reciba un número ingresado por el usuario y averigüe si el número es capicúa o no (Por ejemplo: 12321). Nota: recordar el uso del MOD y el Trunc. No podemos transformar el número a cadena para realizar el ejercicio.

9. El número de combinaciones de m elementos tomados de n es:

$$\binom{m}{n} = \frac{m!}{n!(m-n)!}$$

Diseñar una función que permita calcular el número combinatorio de  $\binom{m}{n}$

Nota: n debe ser mayor a 0 y menor que m.

## Procedimientos:

1. Crea un procedimiento “convertirEspaciado”, que reciba como argumento un texto y muestre una cadena con un espacio adicional tras cada letra. Por ejemplo, “Hola, tú” devolverá “Hola, tú “. Crea un programa principal donde se use dicho procedimiento.
2. Realizar un subproceso que reciba una letra y muestre un mensaje si esa letra esta entre las letras “M” y “T”.
3. Crear un programa que dibuje una escalera de números, donde cada línea de números comience en uno y termine en el número de la línea. Solicitar la altura de la escalera al usuario al comenzar. Ejemplo: si se ingresa el número 3.

1

12

123

4. Realice un algoritmo que solicite al usuario una fecha y muestre por pantalla la fecha anterior. Para ello se deberá utilizar un procedimiento llamado `diaAnterior` que reciba una fecha representada a través de tres enteros día, mes y año, y retorne la fecha anterior. Puede asumir que día, mes y año representan una fecha válida. Realice pruebas de escritorio para los valores día=5, mes=10, año=2012 y para día=1, mes=3, año=2004.
5. Diseñar un procedimiento que reciba una frase, y el programa remueva todas las vocales repetidas. Al final el procedimiento mostrará la frase final.

### Recursión:

6. Crear un programa que calcule la suma de los enteros positivos pares desde N hasta 2. Chequear que si N es impar se muestre un mensaje de error.
7. Implemente de forma recursiva una función que le dé la vuelta a una cadena de caracteres. NOTA: Si la cadena es un palíndromo, la cadena y su inversa coincidirán
8. Realice nuevamente un programa que calcule la función de Fibonacci, pero esta vez de manera recursiva.
9. Escribir un programa que calcule el máximo común divisor (MCD) de dos enteros positivos. Si  $M \geq N$  una función recursiva para MCD es:  
$$\text{MCD} = M, \text{ si } N = 0$$
$$\text{MCD} = \text{MCD}(N, M \bmod N), \text{ si } N \neq 0$$
  
El programa le debe permitir al usuario ingresar los valores para M y N. Una función recursiva es entonces llamada para calcular el MCD. El programa debe imprimir el valor para el MCD.