**A13 - Programación modular**

1. Con programación modular, realiza que dado un número (por el teclado) determine si es par o impar.
2. Con programación modular, determina si un número es negativo, positivo o neutro.
3. Desarrolla un programa modular que permita leer un número entero positivo del teclado y dependiendo del valor capturado despliegue en la pantalla el día de la semana al que le corresponde dicho número, es decir si está en un rango del 1 al 7, el 1 corresponde al Domingo, el 2 corresponde al Lunes, etc.
4. Con programación modular realiza que dado un número entero a multiplicar y la cantidad de iteraciones (ingresados por el teclado), calcula su tabla de multiplicar y se muestra en la consola el producto desde el número 1 hasta el número señalado.
5. Elabora un programa modular que permita ingresar números enteros, el programa finalizará cuando se ingrese un número impar

.

1. Dado un número entero por el teclado, determina si es un primo. Realizar mediante programación modular y el método de ‘*Fuerza bruta*’ es decir, si recibes por ejemplo un 7 deberá validar que ningún número diferente a 1 o a sí mismo. Si un número lo divide entonces se cambia una bandera. Al final pregunta por el estado de la bandera.
2. Dado un número entero por el teclado, determina si es un primo. Realizar mediante programación modular y la siguiente optimización es decir, si recibes por ejemplo un 7 deberá validar que ningún número diferente a 1 o a sí mismo. Si un número lo divide entonces se sale inmediatamente de la función (módulo).
3. Realiza con programación modular que dado un número entero por el teclado, determina cuántos divisores tiene el número. No incluir a los divisores 1 y a sí mismo.
4. Realiza con programación modular que dado un número entero por el teclado, muestra cuáles divisores tiene el número. No incluir a los divisores 1 y a sí mismo.

**NUEVOS PROBLEMAS**

**Problema 1:**

Dado los siguientes números 4, 12, 5, 23, 6, 7 realiza las llamadas requeridas a un procedimiento para construir su tabla de multiplicar requerido, se inicia en 1 y termina en 12. Bajo la forma:

n \* 1 = n

n \* 2 = 2n

n \* 3 = 3n

…

n \* 12 = 12n

Deja al menos un salto de línea entre cada tabla mostrada.

**Problema 2:**

Elabora un programa que pida un número entero al usuario y realizando llamada a una función se calcule su factorial (*n!*). El programa continúa solicitando números al usuario y termina cuando el usuario indica su finalización.

**Problema 3:**

Dada la siguiente sucesión,

*f0 = 0*

*f1 = 1*

*fn = fn-1 + fn-2*

Elabora un programa que pida un número de iteraciones a un usuario y realizando llamada a una función se calcule el fibonacci resultante (*fn*) (un solo número). El programa continúa en un ciclo solicitando números y termina cuando el usuario indica su finalización.

**Problema 4:**

Dada la siguiente sucesión,

*f0 = 0*

*f1 = 1*

*fn = fn-1 + fn-2*

Elabora un programa que pida un número de iteraciones a un usuario y realizando llamada a una función se muestre la serie de fibonacci resultante (*fn*). El programa continúa en un ciclo solicitando números y termina cuando el usuario indica su finalización.