INSTITUTO TECNOLÓGICO AUTÓNOMO DE MÉXICO

Organización y Programación de Computadoras

Tarea BL

Grupo de Trabajo

Cuatro Cuarentenas

Integrantes

América Castrejón – 166414

Renata Monsalve– 176371

Francisco Altamirano – 163314

Rodrigo Castillo — 168114

Fecha (s) de elaboración

28/10/2020

Estos cuatro (4) ejercicios se realizarán en equipo de trabajo.

Deben usar programación estructurada (macro directivas) en la solución de estas preguntas problemas.

El entregable, en archivo “.zip”, deberán subirlo a Canvas a más tardar el lunes 26 de octubre, antes de las 23:30 hs, incluyendo:

* el archivo reporte (.docx), donde en cada pregunta-problema deberán contestar la respuesta correspondente, además de que para las tres primeras preguntas problemas deben incluir:
  + un “pantallazo” de la consola de ejecución, y
  + **un esqueleto de estructuras algorítmicas, en C, C++, Java, correspondientes a su programa, con las condiciones Booleanas respectivas.**
* los tres (3) archivos de los programas “.asm”, con los nombres de archivo declarados \_\_\_\_\_\_\_\_\_.
* La cuarta pregunta es sobre hacer resúmenes del libro de CODE.

**PREGUNTAS PROBLEMAS.**

1. Elabore el programa ensamblador “\_\_\_\_\_\_\_.asm”, que dado un valor n entero, calcule e imprima la siguiente serie sumatoria de n términos:

-1 + 3 - 6 + 10 - 15 + 21 – 28 + continúa

donde n es un dato entero >= 1.

Deberán revisar que la N sea correcta, en caso contrario deberá volver a pedir el dato, imprimiendo antes un texto de “ERROR“, y el valor entero tecleado.

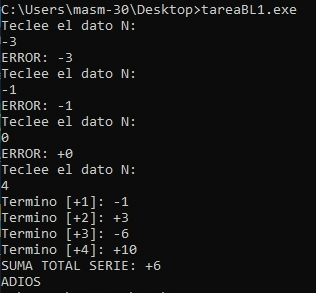
Descubra el paso correspondiente entre términos, explíquelo: \_\_\_\_\_\_\_\_\_\_.

Deberán imprimir los n términos, así como la suma resultante de la serie, con textos adecuados, usando los procedimientos de librería vistos en clase.

El siguiente recuadro despliega el caso de N=4, después de un error.

|  |
| --- |
| Teclee el dato N: -3  ERROR: -3  Teclee el dato N: 4    Termino [+1]: -1  Termino [+2]: +3  Termino [+3]: -6  Termino [+4]: +10  SUMA TOTAL SERIE: +6  ADIOS. |

HINT: Para cambio de signo en los términos, puede usar IMUL, y / o NEG.



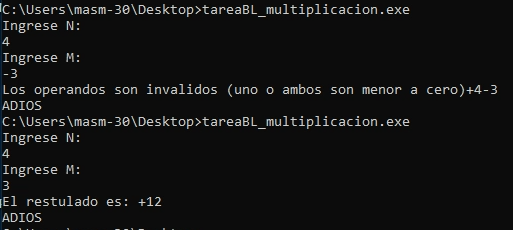
1. Elabore el programa ensamblador “\_\_\_\_\_\_\_.asm”, y revise que los datos dados a m y n sean m ³ 0 y n ³ 0, (ambos datos enteros) para calcular el producto m\*n sin utilizar la operación de multiplicación. En caso de no cumplirse la condición anterior el programa deberá terminar imprimiendo un texto de error e imprimiendo los valores de m y n tecleados inicialmente.

Haga los movimientos necesarios para que independientemente de cómo se hayan teclados los datos, m<=n se cumpla, y calcule el producto haciendo repetir m veces n, la forma más óptima.

Deberá imprimir el producto resultante, con el texto correspondiente, como se muestra en las dos alternativas de despliegue, según sea el caso.

|  |
| --- |
| Teclee el dato M: 4  Teclee el dato N: -3  ERROR: +4, -3    ADIOS. |

|  |
| --- |
| Teclee el dato M: 4  Teclee el dato N: 3    PRODUCTO (+3)\*(+4): +12  ADIOS. |



1. Elabore el programa ensamblador “\_\_\_\_\_\_\_.asm”, que calcule factorial de n (n!), recordando que para n! con:

· n < 0, n! no está definido,

· n = 0, n! = 1, y

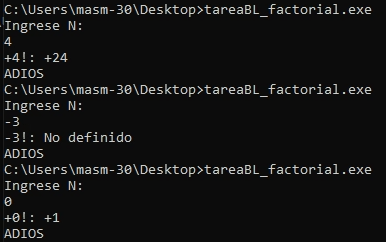
· n > 0, n! está definido como 1\*2\*3\*4\*…\*(n-2)\*(n-1)\*n

Despliegue según corresponda:

|  |
| --- |
| Teclee el dato N: 4    (+4!): +24  ADIOS. |

|  |
| --- |
| Teclee el dato N: -3    (-3!): No definido  ADIOS. |

|  |
| --- |
| Teclee el dato N: 0    (+0!): +1  ADIOS. |



1. Haga un resumen de los capítulos 8 (Alternative to Ten, p54) y 9 (Bit by Bit by Bit, p69), del libro de CODE.

Por qué el autor, en el capítulo 9, ¿empieza refiriéndose a Tony Orlando (cantante del grupo Dawn)?

Como el nombre lo explica, el capítulo 8 explica las diferentes bases de números para poder contar. Comienza diciendo que el sistema numérico que se basa en el número 10 (decimal), este se crea por la única razón de que tenemos 10 dedos y por eso se nos facilita contar. Sin embargo, existen otras alternativas a este sistema numérico. Para evitar confusiones utilizamos un subíndice para indicar el sistema de numeración que estamos utilizando por ejemplo: .

El primer sistema que explica es el octal. Una de las principales diferencias es que cada posición en un número octal es un dígito que se multiplica por una potencia de ocho. Por ejemplo para convertir un número octal a decimal lo podemos hacer de la siguiente forma:

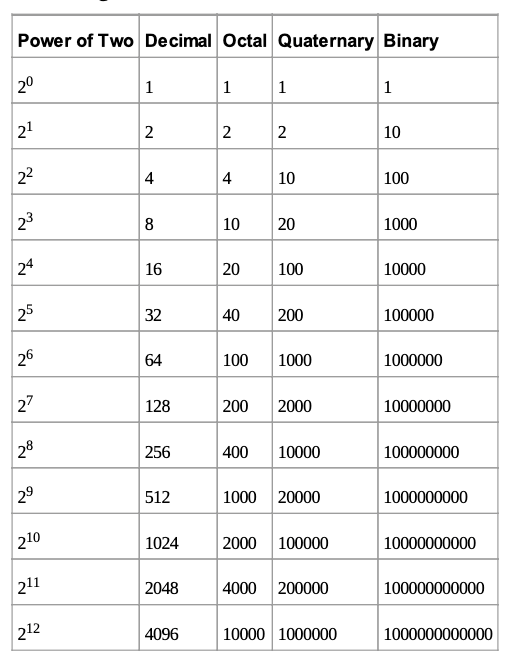
.

El autor menciona que existen otras opciones como la denominada “nice round number”, generalmente nos referimos a este término cuando tiene dos ceros al final. En el caso de números octales, dos ceros al final significa que el número es un múltiplo de que equivale a .

En el caso del sistema cuaternario, cada posición del número corresponde esta vez a una potencia de cuatro:

.

De forma parecida, en el sistema binario cada posición corresponde a una potencia de dos.

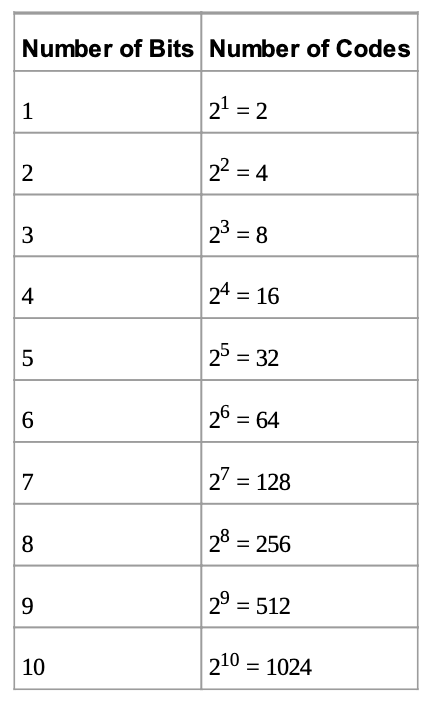


Una desventaja de los números binarios es que pueden volverse muy largos. Para que se vuelva más legible, se acostumbra a separar cada cuatro dígitos binarios con un guión o con espacios.

Al reducir nuestro sistema a sólo dos dígitos lo hemos simplificado lo más posible. El capítulo 9 se dedica a demostrar toda la información que podemos obtener al utilizar únicamente el 0 y el 1.

Un ejemplo que se presenta en el libro es con la canción de Tony Orlando "Tie a Yellow Ribbon Round the OleOak Tree" pues a pesar de que los sentimientos suelen ser complejos, todo lo que el hombre quería era un sí o no. Quería que una cinta amarilla atada alrededor del árbol significaba sí y la ausencia fuera un no. A lo largo del capítulo se muestran diferentes ejemplos para mostrar que se puede elegir entre muchas formas de decir sí o no, no necesita una oraciones ni palabras. Todo lo que se necesita es un 0 y un 1.

La palabra bit en el contexto del sistema binario se refiere a una pequeña porción. Sin embargo con la creación de las computadoras, el bit se volvió el concepto para referirse al bloque de construcción básico de información. Pero como un bit representa la menor cantidad de información posible, se puede transmitir información más compleja con varios bits. En general, el número de valores que se pueden representar es igual a 2 elevado a la potencia del número de bits.



Algunos ejemplos de donde se utilizan el código binario es el código universal de producto donde las líneas verticales de color negro se pueden representar con un 1 y las blancas con un 0. Una vez que se representan las líneas en binario se puede pasar a decimal y obtener un número de serie. Otro ejemplo es el código morse donde un punto se representa con 1, una línea con 11 un espacio con 0 y un espacio entre palabras con 00.

FIN