INSTITUTO TECNOLÓGICO AUTÓNOMO DE MÉXICO

Organización y Programación de Computadoras

Tarea BG

Grupo de Trabajo

Cuatro Cuarentenas

Integrantes

América Castrejón – 166414

Renata Monsalve– 176371

Francisco Altamirano – 163314

Rodrigo Castillo — 168114

Fecha (s) de elaboración

19/10/2020

APLICACIÓN DE LAS INSTRUCCIONES DEL MATERIAL matBG.

Esta tarea es a nivel grupo de trabajo. No olviden incluir la carátula al principio. Para cada uno de los dos ejercicios, de programa ensamblador, deberán incluir un “pantallazo” (screenshot) del despliegue de la ejecución. En el caso de la tercera pregunta sólo deberá responderla. Además, en el mismo archivo .zip, deben incluir este reporte (Word), con la respuesta del ejercicio 3) y los dos archivos fuente (.asm).

Los procedimientos de Irvine, algunos los puede encontrar en matBD2.pptx y todos en el capítulo 5 del libro.

1. Elabore un programa MASM (.asm) donde usted implementará en ensamblador la siguiente operación de asignación con expresión aritmética, expresada en algún lenguaje de alto nivel.

|  |
| --- |
| **R = -A \* 9 – (B / D + 1) + 100** |

Donde los contenidos de A, B, C y R se encuentran definidos variables-etiquetas de tipo SDWORD, en .DATA. **A** deberá tener un valor inicial de 7, **B** sin valor inicial, **D** con valor inicial de -15 y **R** sin valor inicial.

El contenido de **B** deberá ser leído del teclado, como valor entero decimal signado, con el procedimiento **ReadInt**. Deberá imprimir un mensaje, en la consola, que pida el dato (Dato:) para **B**; el valor decimal entero signado. Ver la Figura 1.

El resultado en **R** será impreso con un mensaje adecuado (El resultado R=), con su contenido en formato entero decimal signado haciendo uso de la función **WriteInt**. El resultado aparecerá entre líneas en blanco como se ve en la Figura 1. Además imprima el mismo **R**, pero su contenido en formato hexadecimal.

Al final haga un vaciado, por consola, del contenido de memoria (**DumpMem**) del segmento .DATA (A, B, C y R), apareciendo después del resultado de **Rh**. Además agregue otra línea en blanco después del vaciado.

|  |
| --- |
| Dato: *valor decimal entero signado*    El resultado R= *valor entero decimal signado*  El resultado Rh= *valor hexadecimal*  *vaciado de memoria . . .*  *vaciado de memoria . . .*  HASTA LA VISTA |

Figura 1

**Antes de programar se realizó la jerarquía de operandos**

**{-A} neg -**

**{{-A}\*9} mul -**

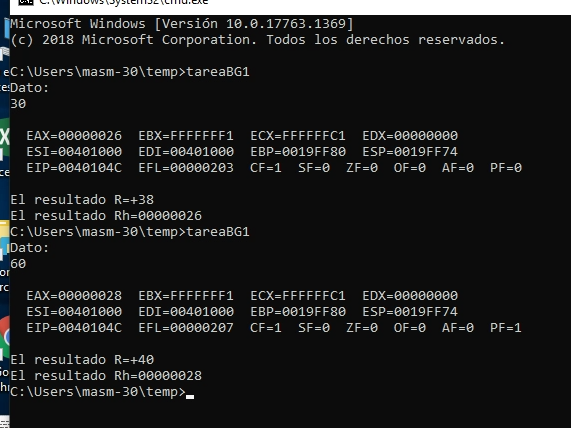
**{B/D} div -**

**{{B/D}+1} add --**

**{{{-A}\*9}-{{B/D}+1}} sub --**

**{{{{-A}\*9}-{{B/D}+1}}+100} add --**

**Las pruebas se corrieron con múltiplos de 15 para evitar floating y facilitarnos el cómputo de la respuesta (mental) para cotejar resultados.**

****

1. Dado el siguiente segmento de DATA:

|  |
| --- |
| .DATA  Svector WORD 2002h, 4004h, 6006h, 8008h  Cvector SWORD -2, -4, -6, -8 |

Desarrolle un programa “.asm” para imprimir:

Primero: la suma, entre sí, del último elemento de cada vector, **Svector** y **Cvector**.

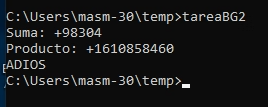
Segundo: el producto de los penúltimos elementos de cada véctor. El resultado es ¿con signo o sin signo?

Para desplegar la salida haga use, entre otros procedimientos de WriteString y WriteInt.

El aspecto de la salida es el siguiente (Figura 2):

|  |
| --- |
| Suma: *valor decimal signado*  Producto: *valor decimal signado*  ADIOS. |

Figura 2



1. Haga un resumen de los capítulos 6 (Telefraphs and Relays, p40) y 7 (Our Ten Digits, p47), del libro de CODE.

**El sexto capítulo narra la historia del telégrafo. Tras una breve introducción sobre el contexto que lo vio nacer y la biografía de su autor, Finley Morse, explica las limitaciones de la comunicación de la época: se limitaba a los alcances de la vista para ser inmediata. El libro explica la física detrás del telégrafo, su diseño y la forma en la que se fue adaptando desde su prototipo inicial para no necesitar de circuitos cerrados ni transcriptores humanos que reciban y transcriban un mensaje (dado que los cables no podían extenderse sobre una distancia ilimitada).**

**El séptimo capítulo habla del desarrollo de sistemas numéricos. Tras mencionar que al contar elementos es más fácil hacer una raya por cada unidad, en vez de dibujar de nuevo todo el elemento, plantea estas rayas como la base de los símbolos con los que después se significaron unidades, decenas, centenas, etc. Sin embargo, la mayoría de los modelos numéricos tienen un símbolo para el 10, mientras que el arábigo contaba con el 0. Con ello, al llegar al 10 no se usaba un símbolo extra sino que se moduló el conteo, desplazando a la izquierda el 1, al completar una centena se desplaza nuevamente una “casilla” a la izquierda y se agrega nuevamente un 0. Esta partición nos permite hacer operaciones aritméticas, por ejemplo, 245 es 200+40+5; cada una de estas cifras está multiplicada por una potencia de la base (10 ya que es un sistema decimal). No se necesita un símbolo nuevo para cada una porque se recorre simplemente a la izquierda y eso le permite escalar indefinidamente la cifra. También eso nos permite hacer uso de decimales, pues las cantidades se desplazan hacia la derecha (tras marcar un punto) y entre más se desplace más pequeña se hace la cantidad. Esto también tiene un crecimiento ilimitado que no depende de símbolos específicos para denominar un “ciclo” como sí lo hacen los números romanos. Cierra explicando que lo mejor del sistema es que la base no requiere ser decimal, así, hasta una caricatura con 4 dedos en cada mano (que sienta inclinación a contar en base 8 al usar todos sus dedos, puede simplemente trabajar con potencias de 8 y al completar un ciclo recorrer a la izquierda el 1 y poner un 0.**

Además, responda, cuál elemento discreto, primero en electrónica y ahora en microelectrónica, substituyó al RELAY, haciendo posible que los CPUs existan como ahora los conocemos.

**Dado que al final de la lectura se explica cómo el telégrafo funciona a través de conexiones modulares que replican mensajes como repetidores, podemos asumir que la respuesta que busca es el transistor (elemento discreto ya que funciona como una unidad elemental). Este recibe también una corriente eléctrica y la repite o amplifica. Los microprocesadores modernos integran más de 700 millones de transistores en la actualidad.**

También, cuál cantidad no podían representar los Romanos, en la antigua Roma, con su sistema numérico, cantidad que, si lo podían representar los Mayas, en su sistema numérico, en la antigua Mesoamérica.

**El 0, de hecho no era una carencia de representación, se podría decir que no lo habían definido con ningún caracter porque el concepto de nulo o ninguna cantidad no lo tenían desarrollado.**

Por cierto, que base numérica empleaban los Mayas en su sistema numérico.

**Vigecimal (20) con el 5 como base auxiliar. Representaban los primeros 4 con puntitos y al completarse la quintupla se representa con una raya, por ejemplo dos rayas son 10, dos rayas y tres puntos son 13. Al completarse la base usaban un punto para simbolizar el número de veintenas, por ejemplo un punto y un cero es 20, dos puntos y un cero abajo serían 40, 2 puntos arriba, con 4 puntos en la siguiente línea son 24 y así sucesivamente. Llegar al 100, donde los puntos se volverían 5, completaba la base auxiliar, por lo que se representa con una raya y un cero abajo.**

1. Estudie, de libro de Irvine el capítulo 7 (Integer Arithmetic). No hay que reportar este punto.

FIN