Tarea BC, entrega el lunes 07oct19.

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

Los procedimientos de Irvine, algunos los puede encontrar en matBA2.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

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 los dos vectores **Svector** y **Cvector**. En ambos casos debe usar **DumpRegs** para la impresión de los valores, guardando previamente los cuatro valores en los registros EAX, EBX, ECX y EDX. Tome en cuenta el tamaño de los valores en memoria y los registros.

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

|  |
| --- |
| Svector  EAX= ::::::::::: EBX= …..  ESI= ……………………….  EIP= ………………....OF=..  Cvector  EAX= ::::::::::: EBX= …..  ESI= ……………………….  EIP= ………………....OF=..  Producto: *producto resultante* |

Figura 2

Al final deberá mostrar el producto del primer elemento de Svector por el primer elemento de Cvector. El resultado es ¿con signo o sin signo? El producto hágalo con operandos a 32 bits.

1. Estudie tanto las instrucciones de división de la presentación matBC.pptx así como los procedimientos de Irvine que vienen en marBA2. Estos dos temas vienen totalmente a gran detalle en los capítulos 5 (Procedures) y 7 (Integer Arithmetic).

También estudien los capítulos 7 (Our Ten Digits) y 8 (Alternatives to Ten), del libro de CODE.

FIN