Assignment 3

**3.1.11**

1. Decimal: -35, -35d  
 Hexadecimal: 0DDh  
 Octal: 335o  
 Binary: 11011101

2. No, A5h is not a valid hexadecimal literal since it doesn’t have ‘0’ as a prefix.

3. No, the \* operator doesn’t have a higher precedence than the / operator; their precedence is the  
 same.

4. (105-5)/(-6+16)\*2 MOD 3  
 = (100)/(10) \* 2 MOD 3  
 = 10 \* 2 MOD 3  
 = 20 MOD 3  
 = 2

**3.2.4**

4. The EAX register holds the sum.

5. The EXIT statement halts the program.

**3.3.3**

1. The assembler produces object (.obj) files and listing (.lst) files.

2. True, the linker takes the object files and performs all checks and procedures needed to output the  
 .exe file.

3. True, since the source code needs to be assembled the linker must go through its previous steps to  
 produce the .exe file again.

4. The Loader reads and executes programs.

**3.4.13**

1. label SWORD ?

2. label BYTE ?

3. label SBYTE ?

4. label QWORD ?

5. SDWORD ?

**3.5.5**

3. myArray WORD 20 DUP(?)  
 ArraySize = ($ - myArray)

**3.9.1**

4. Assembler is the name of the program that converts source files into object files so it can be  
 understood by a machine. The object files are then assembled into executable files; thus assembly  
 language is better name than assembler.

5. Big-endian order is ordering bytes from high to low; the most significant byte is stored at the first  
 memory address.  
  
 Little-endian order is ordering the bytes from low to high; the least significant byte will be stored at  
 the first memory address.  
  
 Big-endian order was used before micro-processors due to chip limitations. Little-endian order was  
 then developed since it used simpler logic.

6. A symbolic constant does not reserve any memory space. You would also want to use once when the  
 value of an integer literal may remain constant during runtime.

25. SDWORD

**3.9.2**

4. A negative value cannot be stored with DWORD, but it can with SDWORD. This shows that the  
 programmer needs to be checking for the type of values are being inputted in the correct range.

7. arr1 DWORD 120 DUP (?)

13. str1 BYTE 500 DUP (“Test”)

**3.10**

TITLE Symbolic Integer Constants (symlnt.asm)

INCLUDE Irvine32.Inc

Monday = 1

Tuesday = 2

Wednesday = 3

Thursday = 4

Friday = 5

Saturday = 6

Sunday = 7

main PROC

exit

main ENDP

END main