- [ ] 3.1.11 Section Review, Questions 1, 2, 3, 4

1. Using the value –35, write it as an integer literal in decimal, hexadecimal, octal, and binary formats that are consistent with MASM syntax.

-35d

-00010011b

-23h

-43o

2. *(Yes/No):* Is A5h a valid hexadecimal literal?

yes

3. *(Yes/No):* Does the multiplication operator (\*) have a higher precedence than the division operator (/) in integer expressions?

no

4. Create a single integer expression that uses all the operators from Section 3.1.2. Calculate the value of the expression.

-3 + 100 \* (100 / 10 MOD 2) – 7 = -10

- [ ] 3.2.4 Section Review, Questions 4, 5

4. In the AddTwo program, which register holds the sum?

The eax register.

5. In the AddTwo program, which statement halts the program?

Either the ExitProcess or the end.

- [ ] 3.3.3 Section Review, Questions 1, 2, 3, 4

1. What types of files are produced by the assembler?

An object file and optionally a listing file.

2. *(True/False):* The linker extracts assembled procedures from the link library and inserts them in the executable program.

False

3. *(True/False):* When a program’s source code is modified, it must be assembled and linked again before it can be executed with the changes.

True

4. Which operating system component reads and executes programs?

The loader.

- [ ] 3.4.13 Section Review, Questions 1, 2, 3, 4, 5

1. Create an uninitialized data declaration for a 16-bit signed integer.

num SWORD

2. Create an uninitialized data declaration for an 8-bit unsigned integer.

num BYTE

3. Create an uninitialized data declaration for an 8-bit signed integer.

num SBYTE

4. Create an uninitialized data declaration for a 64-bit integer.

num QWORD

5. Which data type can hold a 32-bit signed integer?

num SDWORD

- [ ] 3.5.5 Section Review Question 3

3. Write a statement that causes the assembler to calculate the number of bytes in the following array, and assign the value to a symbolic constant named **ArraySize**:

myArray WORD 20 DUP(?)

ArraySize = (? – myArray) / 2

- [ ] 3.9.1 Short Answer, Questions 4, 5, 6, 25

4. Explain why the term *assembler language* is not quite correct.

Because there are multiple different assemblers out there, all for different languages.

5. Explain the difference between big endian and little endian. Also, look up the origins of this term on the Web.

Little endian arranges data by the lowest memory address to highest memory address. Big endian arranges data by highest memory address to lowest memory address.

6. Why might you use a symbolic constant rather than an integer literal in your code?

Because a symbolic constant can be altered to affect all that use it in the code easier.

25. Which data directive creates a 32-bit signed integer variable?

SDWORD

- [ ] 3.9.2 Algorithm Workbench, Questions 4, 7, 13

4. Find out if you can declare a variable of type DWORD and assign it a negative value. What does this tell you about the assembler’s type checking?

It does not check if the type is signed or unsigned.

7. Declare an array of 120 uninitialized unsigned doubleword values.

array DWORD 120 DUP (?)

13. Declare a string variable containing the word “TEST” repeated 500 times.

strArray BYTE 500 DUP ("TEST")

- [ ] 3.10 Programming Exercises, Pick any 1 exercise