Introduction to 80x86 Assembly Language and Computer Architecture (2nd ed)
Richard C. Detmer
Corrections as of 4/9/2010 (* means corrected in 2nd printing)

p39, Figure 2.4, third line for RBX, RCX, etc. should read "low-order word BX, CX, DX, SI, DI"

p95, Figure 4.5, first line below "64-bit mode" change R8B to R8W

p100, 2 lines above 64-bit mode, 1st column, change "except AX" to "except EAX"

p102, change CEX to ECX in problem 4.1-1(a) "after" column

p102, change Value to dValue in problem 4.1-1(c) "after" column

p102, change Value to dValue in problem 4.1-1(f) "after" column

p103, change 01 A2 to 00 00 01 A2 in problem 4.1-4(b)

p104, each example is missing a second "before" condition: first example should have "ECX: 00 00 01 A2," second example should also have "ECX: 00 00 01 A2," the third example should have "CX: 4B 35," and the fourth example should have "ECX: 00 00 01 A2."

p105, the second example is missing the "before" condition "word at Value: FF 20" and the last example is missing the before condition "doubleword at Dbl: 00 00 01 00" (the value was omitted). Both of these can have the value indented on the line "word at" or "doubleword at."

p129, line1 : change AX to EAX in (h)

*p187, line 1: change 2 to 1 to give "for position := 1 to nbrElts-1 loop"

p192, 3rd line above Figure 6.2: replace 0x0044FD88 by 0x002BFC88.

p193, 2nd line after Figure 6.3: replace 0044FD94 by 002BFC90.

p193, 6th line after Figure 6.3: replace 0044FD90 by 002BFC94.

p200, line 2 of Figure 6.7 should say "; returns 3*x+7*y"

*p207, 3rd line of problem 1: change ";" to "//" so that the line reads "// return the ..."

p214, 5th line from bottom: remove "NEAR32" from "move PROC NEAR32"

p229, problem 4: change each of the two semicolons in the code to "//" giving

```
// sort nbrArray...
// into increasing...
```

p243, first example: under AX: E2 75, add a 2nd line of before information "CX: A9 D7" This makes the first example look like:

AX: E2 75 CX: A9 D7	and ax,cx	1110 0010 0111 0101 1010 1001 1101 0111 1010 0000 0101 0101	AX SF 1	A0 ZF 0	55

*p243, second example: under DX: E2 75, add a 2nd line of "before" information "word at value: A9 D7"
This makes the 2nd example look like:

DX: E2 75 word at value: A9 D7	`or dx,value`	1110 0010 0111 0101 1010 1001 1101 0111 1110 1011 1111 0111	DX SF 1	EB ZF 0	F7

*p254, in the 2nd line of the first example: change the leading 1011 to 1001, making the entire example look like

word at ace: A9 D7	sal ace,4	1010|1001 1101 0111 ← 1001 1101 0111 **0000**	ace SF 1 CF 0	9D ZF 0 OF ?	70

*p277, line above the figure: change "discussed on the next page" to "discussed on page 283"

*p305, last line of middle paragraph: remove the word "or" to change the parenthetical sentence to read "(There is, however, an instruction to transfer the status word to AX.)"

p316, change the last instruction mnemonic in Figure 9.6 from "fsubpr" to "fsubrp"

p342, the algorithm in problem 2 should read

```
root := 1.0;
repeat
        oldRoot := root;
        root := (2.0*root + x/(root*root))/3.0;
until (|root - oldRoot| < smallValue);
```

*p367, last line of page: delete the entire line "unsigned check: 32766+2=32768"

p368, Exercises 2.1 (page 32), 2: replace 2×3^{30} by 2×2^{30}

p369, 3.3-10: this answer should be
"3C 3E 3C 3E 3C 3E 3C 3E 3C 3E (shown as 00000005 [3E3C], but the assembler displays the bytes backwards)

p370, 4.3.2(f): this answer should be "opcode F7, 2 bytes"

p373, 6.1.2(a): should have "ESP: 06 00 0F F8" as the top line on the left