Unit 2 errata

Whilst every effort was made to correct errors before the book was published some errors were not corrected. These were not spotted by our two reviewers and a proof reading or if they were, their correction was unintentionally overlooked. These errors will be corrected in the next edition of the textbook. Our apologies if these errors have caused confusion.

Page 63 Table 5.4.4.3 should be

|  |  |  |
| --- | --- | --- |
| Fractional  part, F | R | Digit, D |
| 0.1 | 0.1 x 2 = 0.2 | 0 |
| 0.2 | 0.2 x 2 = 0.4 | 0 |
| 0.4 | 0.4 x 2 = 0.8 | 0 |
| 0.8 | 0.8 x 2 = 1.6 | 1 |
| 0.6 | 0.6 x 2 = 1.2 | 1 |
| 0.2 | 0.2 x 2 = 0.4 | 0 |
| 0.4 | 0.4 x 2 = 0.8 | 0 |
| 0.8 | 0.8 x 2 = 1.6 | 1 |
| 0.6 | 0.6 x 2 = 1.2 | 1 |
| 0.2 | (Previous 4 steps now repeat) |  |

Table 5.4.4.3 Repeating binary pattern for which F ≠ 0 and F ≠ OrigF.

Page 82: Table 7.3.3.5 contains an error in the Opcode column, B <condition> <lable> row

It should have been as follows

011101

011110

011111

100000

Page 188: “...first convert the plaintext letters to their ...” should have been “...first convert the ciphertext letters to their ...”

Page 191: 2, 14, 26, 36 should have been 2, 14, 26, 38 as in Figure 6.10.10

Page 206: In Table 6.10.3 the row 1 1 1 should have been 1 0 1 and the last row 1 1 0. See page 235, Figure 6.4.1.14 for correct truth table.

Page 248 Figure 6.5.1.2 In the Truth table for A OR NOT A, the two values in the output column should both be one.

Page 495: The following was accidently hidden behind Question 8 –

**Car**(CarId, CarModel, *DriverId*, …..)

Page 567 Python code:

We first define a function double which takes two arguments, afunction and x. The argument afunction is a function type and the argument x has type integer. The function double returns 2\*afunction(x).

...................................

def double (afunction, x):

return 2\*afunction(x)

....................................

def cube (x):

return x\*x\*x

Now we can use double, square and cube as follows

double(square, 4)

The function call double(square, 4) returns 32.

double (cube, 4)

The function call double(cube, 4) returns 128.

Page 590 foldl

With foldl (^) 2 [1,2,3] in WinGHCi, the accumulator is 2. The list is folded from the left as follows

accumulator ^ list value

2^1 = 2

2^2 = 4

4^3 = 64