1. Suppose an 8-bit data word stored in memory is 11000010. Using the Hamming algorithm, determine what check bits would be stored in memory with the data word. Show all work.

Check bits are in positions 1, 2, 4, 8

C1= D1 xor D2 xor D4 xor D5 xor D7 = 1 xor 1 xor 0 xor 0 xor 1 = 1

C2= D1 xor D3 xor D4 xor D6 xor D7 = 1 xor 0 xor 0 xor 0 xor 1 = 0

C4= D2 xor D3 xor D4 xor D8 = 1 xor 0 xor 0 xor 0 = 1

C8= D5 xor D6 xor D7 xor D8 = 0 xor 0 xor 1 xor 0 = 1

It would store the check bits: 1011

1. For the 8-bit word 00111001, the check bits stored with it would be 0111. Suppose when the word is read from memory, the check bits are calculated to be 1101. What is the data word that was read from memory?

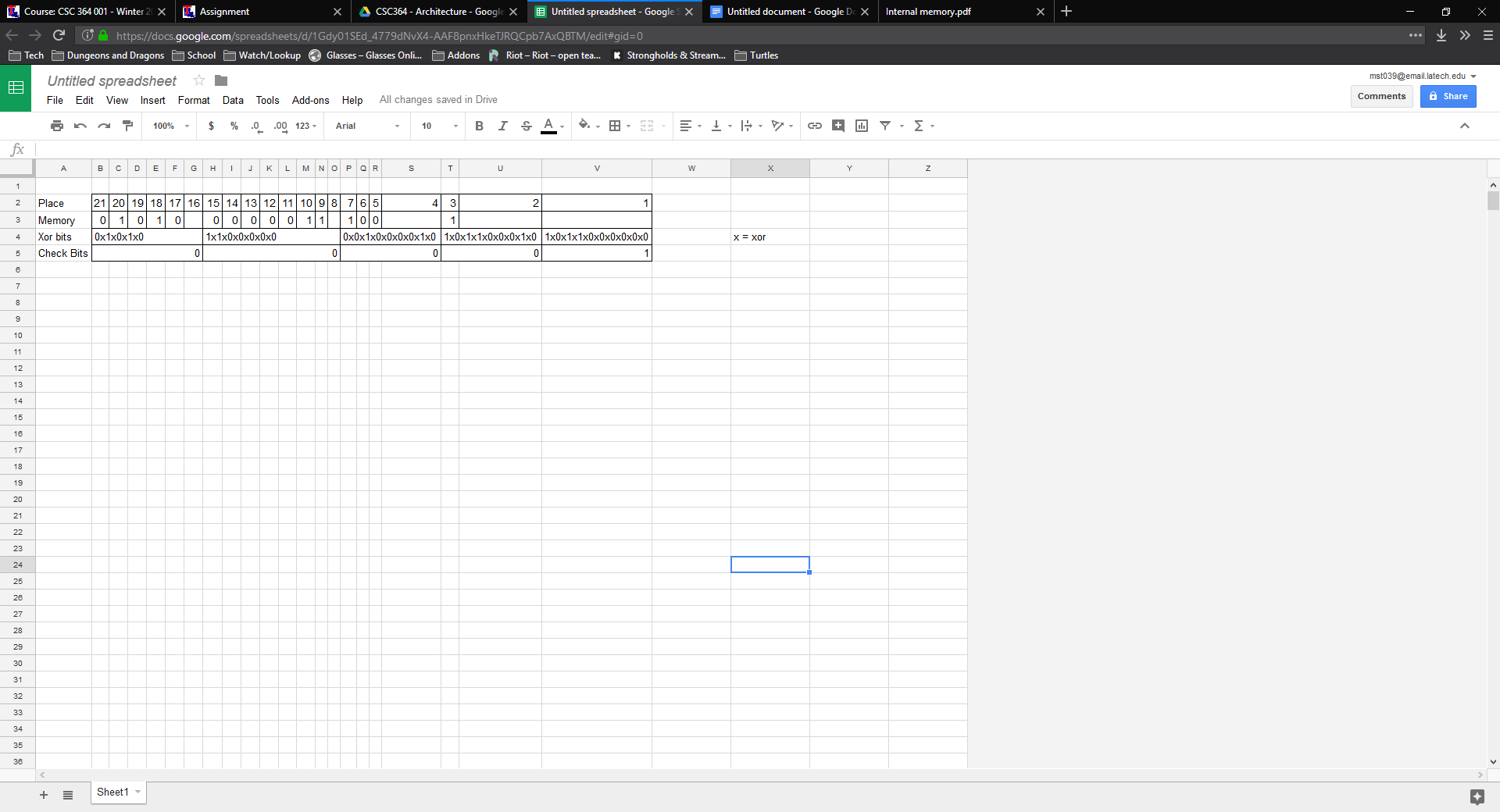
|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Place | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 |
| Memory | 0 | 0 | 1 | 1 | **0** | 1 | 0 | 0 | **1** | 1 | **1** | **1** |

0111 xor 1101 = 1010 = 10 (decimal)

That means there is an error in bit 10

So taking the data word and flipping the 1 in position 10 gives the data word: 00011001

1. Develop an SEC (Single Error Correction) code for a 16-bit data word. Generate the code for the data word 0101 0000 0011 1001. Show that the code will correctly identify an error in **data bit** 4.



So if there was an error in data bit 4, that would be in place 7(bolded)

If place 7 had an error and was flipped to a 0 the xor bits would be

0 xor 0 xor 0 xor 0 xor 0 xor 0 xor 0 xor 1 xor 0 = 1

This makes the error check bits 00101

Xor with the original check bits 00001 give 00100 = 4

Showing that there is an error in data bit 4 (place 7)