COMPSCI 4CR3 - Assignment 2

1. An encryption function $e(k, x)$, where k is the key and x is the message, is said to be linear if $e(k, x + y) = e(k, x) + e(k, y)$ for all keys k and all messages x, y.

   (a) (15 points) Explain why a secure encryption function must not be linear; provide an example scenario where Trudy could dangerously exploit this linearity.

   (b) (15 points) Recall that the S-box in AES is a function

   $$f : \{0, 1, \ldots, 255\} \to \{0, 1, \ldots, 255\}.$$

   The actual S-box is given by the following table.

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	63	7C	77	7B	F2	6B	6F	C5	30	1	67	2B	FE	D7	AB	76
1	CA	82	C9	7D	FA	59	47	F0	AD	D4	A2	AF	9C	A4	72	C0
2	B7	FD	93	26	36	3F	F7	CC	34	A5	E5	F1	71	D8	31	15
3	4	C7	23	C3	18	96	5	9A	7	12	80	E2	EB	27	B2	75
4	9	83	2C	1A	1B	6E	5A	A0	52	3B	D6	B3	29	E3	2F	84
5	53	D1	0	ED	20	FC	B1	5B	6A	CB	BE	39	4A	4C	58	CF
6	D0	EF	AA	FB	43	4D	33	85	45	F9	2	7F	50	3C	9F	A8
7	51	A3	40	8F	92	9D	38	F5	BC	B6	DA	21	10	FF	F3	D2
8	CD	0C	13	EC	5F	97	44	17	C4	A7	7E	3D	64	5D	19	73
9	60	81	4F	DC	22	2A	90	88	46	EE	B8	14	DE	5E	0B	DB
A	E0	32	3A	0A	49	6	24	5C	C2	D3	AC	62	91	95	E4	79
B	E7	C8	37	6D	8D	D5	4E	A9	6C	56	F4	EA	65	7A	AE	8
C	BA	78	25	2E	1C	A6	B4	C6	E8	DD	74	1F	4B	BD	8B	8A
D	70	3E	B5	66	48	3	F6	0E	61	35	57	B9	86	C1	1D	9E
E	E1	F8	98	11	69	D9	8E	94	9B	1E	87	E9	CE	55	28	DF
F	8C	A1	89	0D	BF	E6	42	68	41	99	2D	0F	B0	54	BB	16

   For an input (uv), in base 16, the output is located in row u and column v. For example, $f(1E) = 72$. Show that this S-box is not a linear function.

2. (45 points) Suppose we replace the functions in SHA-256 with following functions.

$$\begin{aligned}
\mathrm{Ch}(x, y, z) &= (x \wedge y) \oplus (x \wedge z) \\
\mathrm{Ma}(x, y, z) &= (x \wedge y) \oplus (x \wedge \neg z) \oplus (y \wedge z) \\
\Sigma_0(x) &= (x \gg_R 2) \oplus (x \gg_R 23) \oplus (x \gg_R 12) \\
\Sigma_1(x) &= (x \gg_R 16) \oplus (x \gg_R 21) \oplus (x \gg_R 15) \\
\sigma_0(x) &= (x \gg_R 17) \oplus (x \gg_R 11) \oplus (x \gg_S 13) \\
\sigma_1(x) &= (x \gg_R 7) \oplus (x \gg_R 9) \oplus (x \gg_S 12)
\end{aligned}$$

Compute the hash of the following message using the modified SHA-256:
`The quick brown fox jumped over the lazy dog.`
Rules to follow:

- The input to the hash should be a byte string. For example, in Python, add a `b` so that the input looks like `b'The quick brown fox jumped over the lazy dog.'`
- The output should be a list of bytes. For the hash of the above message, represent the output bytes in base 16 as a string. Example input and output:
  `COMPSCI 4CR3`
  `8b709893e1b5f6008bfa29295ab4dd2fc0cc81cb68e22c5bb6a69a79d57e6db4`
- Include your source code; Python is preferred, but you can use other languages.

3. (25 points) Let $h_1 : \{0,1\}^* \to \{0,1\}^n$ and $h_2 : \{0,1\}^* \to \{0,1\}^n$ be hash functions with output lengths of n bits. Construct a hash function $h : \{0,1\}^* \to \{0,1\}^{2n}$ by concatenating the outputs of h_1 and h_2, that is, $h(x) = h_1(x) \| h_2(x)$ where $\|$ is concatenation. Suppose h_1 is collision resistant but h_2 is not. Can we say that h is collision resistant? Prove your claim.