Homework 4

Problem 1

a.
Block Length: 16 bits
The state array for a simplified AES is similar to the regular version except the state array is divided into a two by two array (rather than a four by four). These state arrays are referred to as Nibbles (a nibble is also half a byte)

b.
$x^4 + x + 1$

c.
I am doing this under the assumption that the shift rows step has already taken place.
$S_{0,0} = x^3 + 1 = 1001 = 9$
$S_{1,0} = x = 0010 = 2$
$S_{0,1} = x = 0010 = 2$
$S_{1,1} = x^3 + 1$
$S = \begin{bmatrix} 9 & 2 \\ 2 & 9 \end{bmatrix}$
$\begin{bmatrix} 1 & 4 \\ 4 & 1 \end{bmatrix} \times \begin{bmatrix} 9 \\ 2 \end{bmatrix} = \begin{bmatrix} 17 \\ 38 \end{bmatrix} \mod x^4 + x + 1 = \begin{bmatrix} 2 \\ 0 \end{bmatrix} = S'_{0,0-1,0}$
$\begin{bmatrix} 1 & 4 \\ 4 & 1 \end{bmatrix} \times \begin{bmatrix} 2 \\ 9 \end{bmatrix} = \begin{bmatrix} 38 \\ 17 \end{bmatrix} \mod x^4 + x + 1 = \begin{bmatrix} 0 \\ 2 \end{bmatrix} = S'_{0,1-1,1}$
The inverse then becomes:
$\begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix}$

Problem 2

a.
Key = 0e 00 71 c9 47 d9 e8 59 1c b7 ad d6 af 7f 67 98
Key Expansion for AES

Key Words	Auxiliary Funciton
W0 = 0e 00 71 c9 W1 = 47 d9 e8 59 W2 = 1c b7 ad d6 W3 = af 7f 67 98	RotWord(W3) = 7f 67 98 af SubWord(X1) = d2 85 46 79 Rcon (1) = 01 00 00 00 y xor Rcon = d3 85 46 79
W4 = W0 $\oplus$ z = dd 85 37 b0 W5 = W4 $\oplus$ W1 = 9a 5c df e9 W6 = W5 $\oplus$ W2 = 86 eb 72 3f W7 = W6 $\oplus$ W3 = 29 94 15 a7	RotWord(W7) = 94 15 a7 29 SubWord(X2) = 22 59 5c a5 Rcon (2) = 02 00 00 00 y xor Rcon = 20 59 5c a5
W8 = W4 $\oplus$ z = fd dc 6b 15 W9 = W8 $\oplus$ W5 = 67 80 b4 fc W10 = W9 $\oplus$ W6 = e1 6b c6 c3 W11 = W10 $\oplus$ W7 = c8 ff d3 64	RotWord(W11) = ff d3 64 c8 SubWord(X3) = 16 66 43 e8 Rcon (3) = 04 00 00 00 y xor Rcon = 12 66 43 e8
W12 = W8 $\oplus$ z = ef ba 28 fd W13 = W12 $\oplus$ W9 = 88 3a 9c 01 W14 = W13 $\oplus$ W10 = 69 51 5a c2 W15 = W14 $\oplus$ W11 = a1 ae 89 a6	RotWord(W15) = ae 89 a6 a1 SubWord(X4) = e4 a7 24 32 Rcon (4) = 08 00 00 00 y xor Rcon = ec a7 24 32

Key Words	Auxiliary Funciton
W16 = W12 ⊕ z = 31 d0 cc f W17 = W16 ⊕ W13 = 8b 27 90 ce W18 = W17 ⊕ W14 = e2 76 ca 0c W19 = W18 ⊕ W15 = 43 d8 43 aa	RotWord(W19) = d8 43 aa 43 SubWord(X5) = 61 1a ac 1a Rcon (5) = 10 00 00 00 y xor Rcon = 71 1a ac 1a
W20 = W16 ⊕ z = 72 07 a0 d5 W21 = W20 ⊕ W17 = f9 20 30 1b W22 = W21 ⊕ W18 = 1b 56 fa 17 W23 = W22 ⊕ W19 = 58 8e b9 bd	RotWord(W23) = 8e b9 bd 58 SubWord(X6) = 19 56 7a 6a Rcon (6) = 20 00 00 00 y xor Rcon = 39 56 7a 6a
W24 = W20 ⊕ z = 4b 51 da bf W25 = W24 ⊕ W21 = b2 71 ea a4 W26 = W25 ⊕ W22 = a9 27 10 b3 W27 = W26 ⊕ W23 = f1 a9 a9 0e	RotWord(W27) = a9 a9 0e f1 SubWord(X7) = d3 d3 ab a1 Rcon (7) = 40 00 00 00 y xor Rcon = 93 d3 ab a1
W28 = W24 ⊕ z = d8 82 71 1e W29 = W28 ⊕ W25 = 6a f3 9b ba W30 = W29 ⊕ W26 = c3 d4 8b 09 W31 = W30 ⊕ W27 = 32 7d 22 07	RotWord(W31) = 7d 22 07 32 SubWord(X8) = ff 93 c5 23 Rcon (8) = 80 00 00 00 y xor Rcon = 7f 93 c5 23
W32 = W28 ⊕ z = a7 11 b4 3d W33 = W32 ⊕ W29 = cd e2 2f 87 W34 = W33 ⊕ W30 = e3 6a 48 e W35 = W34 ⊕ W31 = 3c 4b 86 89	RotWord(W35) = 4b 86 89 3c SubWord(X9) = b3 44 a7 eb Rcon (9) = 1B 00 00 00 y xor Rcon = a8 44 a7 eb
W36 = W32 ⊕ z = f5 51 3d 6 W37 = W36 ⊕ W33 = c2 b7 3c 51 W38 = W37 ⊕ W34 = cc 81 98 df W39 = W38 ⊕ W35 = f0 ca 1e 56	RotWord(W39) = ca 1e 56 f0 SubWord(X10) = 74 72 b1 8c Rcon (10) = 36 00 00 00 y xor Rcon = 42 72 b1 8c
W40 = W36 ⊕ z = 4d 27 a2 5a W41 = W40 ⊕ W37 = 8f 90 9e 0b W42 = W41 ⊕ W38 = 43 11 06 d4 W43 = W42 ⊕ W39 = b3 db 18 82	

b.
Disclaimer: There were a ton of these and I decided that you could probably figure it out without it being the exact same format even if it was required to be that format. I believe in you Simeon.

Start of Round	After SubBytes	After ShiftRows	After MixColumns	Round Key
1321456789abcdeffedcba9876543210				13 21 45 67 89 ab cd ef fe dc ba 98 76 54 32 10
1d 21 34 ae ce 72 25 b6 e2 6b 17 4e d9 2b 55 88	a4 fd 18 e4 8b 40 3f 4e 98 7f f0 2f 35 f1 fc c4	a4 40 f0 c4 8b 7f fc e4 98 f1 18 4e 35 fd 3f 2f	a7 eb 48 d4 94 8e 20 d6 75 07 8b c6 66 ba c7 c3	a7 eb 48 d4 94 8e 20 d6 75 07 8b c6 66 ba c7 c3
7a 6e 7f 64 0e d2 ff 3f f3 ec f9 f9 4f 2e d2 64	da 9f d2 43 ab b5 16 75 0d ce 99 99 84 31 b5 43	da b5 99 43 ab ce b5 43 0d 31 d2 75 84 9f 16 99	b1 58 83 df f2 ab d1 1b ee 77 1c 1e 26 02 87 37	b1 58 83 df f2 ab d1 1b ee 77 1c 1e 26 02 87 37
4c 84 e8 ca 95 2b 65 e7 0f 1c da dd ee fd 54 53	29 5f 9b 74 2a f1 4d 94 76 9c 57 c1 28 54 20 ed	29 f1 57 ed 2a 9c 20 74 76 54 9b 94 28 5f 4d c1	e0 c4 5a 1c bf 1d 6a 2a 1f fc a8 66 3d 80 b5 f3	e0 c4 5a 1c bf 1d 6a 2a 1f fc a8 66 3d 80 b5 f3
0f 7e 72 e1 37 27 f6 2b 76 ad f2 a4 9c 2e 3c 55	76 f3 40 f8 9a cc 42 f1 38 95 89 49 de 31 eb fc	76 cc 89 fc 9a 95 eb f8 38 31 40 f1 de f3 42 49	d6 89 ac 3c 98 75 d1 20 92 6b 81 c0 a2 ac 72 5a	d6 89 ac 3c 98 75 d1 20 92 6b 81 c0 a2 ac 72 5a

Start of Round	After SubBytes	After ShiftRows	After MixColumns	Round Key
d5 94 a0 f3 13 52 41 ee 70 1d 4b cc e1 74 31 f0	03 22 e0 0d 7d 00 83 28 51 a4 b3 4b f8 92 c7 8c	03 00 b3 8c 7d a4 c7 0d 51 92 e0 28 f8 22 83 4b	39 41 f1 b5 c7 71 5b fe c7 7d 60 d1 45 69 1a 24	39 41 f1 b5 c7 71 5b fe c7 7d 60 d1 45 69 1a 24
4b 46 51 60 3e 51 6b e5 dc 2b 9a c6 1d e7 a3 99	b3 5a d1 d0 b2 d1 7f d9 86 f1 b8 b4 a4 94 0a ee	b3 d1 b8 ee b2 f1 0a d0 86 94 d1 d9 a4 5a 7f b4	43 37 20 60 ad 85 3c 8d b8 04 db 7d 76 25 c7 a1	43 37 20 60 ad 85 3c 8d b8 04 db 7d 76 25 c7 a1
08 66 fa df 1f f4 d6 29 11 23 cb ce 87 8c 6e af	30 33 2d 9e c0 bf f6 a5 82 26 1f 8b 17 64 9f 79	30 bf 1f 79 c0 26 9f 9e 82 64 2d a5 17 33 f6 8b	dc 0d 3a 02 f0 a8 7a c5 3b 98 48 85 06 fb 55 f1	dc 0d 3a 02 f0 a8 7a c5 3b 98 48 85 06 fb 55 f1
04 8f 4b 1c 9a 5b e1 7f f8 4c c3 8c 34 86 77 f6	f2 73 b3 9c b8 39 f8 d2 41 29 2e 64 18 44 f5 42	f2 39 2e 42 b8 29 f5 9c 41 44 b3 d2 18 73 f8 64	d8 b0 51 9e 79 72 df 2c 2f d5 15 8b 39 89 2c 6b	d8 b0 51 9e 79 72 df 2c 2f d5 15 8b 39 89 2c 6b
7f a1 e5 a3 b4 90 f0 ab 21 e3 b1 05 05 c2 aa e2	d2 32 d9 0a 8d 60 8c 62 fd 11 c8 6b 6b 25 ac 98	d2 60 c8 98 8d 11 ac 0a fd 25 d9 62 6b 32 8c 6b	4f c9 8a ee 94 4a c1 25 35 a5 d7 24 67 eb e7 d5	4f c9 8a ee 94 4a c1 25 35 a5 d7 24 67 eb e7 d5
40 9c 99 38 56 fd fd 74 f9 24 4f fb 97 21 f9 83	09 de ee 07 b1 54 54 92 99 36 84 0f 88 fd 99 ec	09 54 84 ec b1 36 99 07 99 fd ee 92 88 de 54 0f		09 54 84 ec b1 36 99 07 99 fd ee 92 88 de 54 0f

c.
Key = "0f 55 71 c9 47 d9 e8 59 0c b7 ad d6 af 7f 67 98"
PlainText = "22 00 45 67 89 ab cd ef fe dc ba 98 76 54 32 10"

Key Expansion:

Key Words	Auxiliary Functions
W0 = 0f 15 71 c9 W1 = 47 d9 e8 59 W2 = 0c b7 ad d6 W3 = af 7f 67 98	RotWord(W3) = 7f 67 98 af SubWord(X1) = d2 85 46 79 Rcon (1) = 01 00 00 00 y xor Rcon = d3 85 46 79
W4 = W0 ⊕ z = dc 90 37 b0 W5 = W4 ⊕ W1 = 9b 49 df e9 W6 = W5 ⊕ W2 = 97 fe 72 3f W7 = W6 ⊕ W3 = 38 81 15 a7	RotWord(W7) = 81 15 a7 38 SubWord(X2) = 0c 59 5c 07 Rcon (2) = 02 00 00 00 y xor Rcon = e5 95 c0 7
W8 = W4 ⊕ z = d2 c9 6b b7 W9 = W8 ⊕ W5 = 49 80 b4 5e W10 = W9 ⊕ W6 = de 7e c6 61 W11 = W10 ⊕ W7 = e6 ff d3 c6	RotWord(W11) = ff d3 c6 e6 SubWord(X3) = 16 66 b4 8e Rcon (3) = 04 00 00 00 y xor Rcon = 12 66 b4 8e
W12 = W8 ⊕ z = c0 af df 39 W13 = W12 ⊕ W9 = 89 2f 6b 67 W14 = W13 ⊕ W10 = 57 51 ad 06 W15 = W14 ⊕ W11 = b1 ae 7e c0	RotWord(W15) = ae 7e c0 b1 SubWord(X4) = e4 f3 ba c8 Rcon (4) = 08 00 00 00 y xor Rcon = ec f3 ba c8
W16 = W12 ⊕ z = 2c 5c 65 f1 W17 = W16 ⊕ W13 = a5 73 0e 96 W18 = W17 ⊕ W14 = f2 22 a3 90 W19 = W18 ⊕ W15 = 43 8c dd 50	RotWord(W19) = 8c dd 50 43 SubWord(X5) = 64 c1 53 1a Rcon (5) = 10 00 00 00 y xor Rcon = 74 c1 53 1a
W20 = W16 ⊕ z = 58 9d 36 eb W21 = W20 ⊕ W17 = fd ee 38 7d W22 = W21 ⊕ W18 = fc c9 be d W23 = W22 ⊕ W19 = 4c 40 46 bd	RotWord(W23) = 40 46 bd 4c SubWord(X6) = 09 5a 7a 29 Rcon (6) = 20 00 00 00 y xor Rcon = 29 5a 7a 29

Key Words	Auxiliary Functions
W24 = W20 ⊕ z = 71 c7 4c c2 W25 = W24 ⊕ W21 = 8c 29 74 bf W26 = W25 ⊕ W22 = 83 e5 ef 52 W27 = W26 ⊕ W23 = cf a5 a9 ef	RotWord(W27) = a5 a9 ef cf SubWord(X7) = 06 d3 df 8a Rcon (7) = 40 00 00 00 y xor Rcon = 46 d3 df 8a
W28 = W24 ⊕ z = 37 14 93 48 W29 = W28 ⊕ W25 = bb 3d e7 f7 W30 = W29 ⊕ W26 = 38 d8 08 a5 W31 = W30 ⊕ W27 = f7 7d a1 4a	RotWord(W31) = 7d a1 4a f7 SubWord(X8) = ff 32 d6 68 Rcon (8) = 80 00 00 00 y xor Rcon = 7f 32 d6 68
W32 = W28 ⊕ z = 48 26 45 20 W33 = W32 ⊕ W29 = f3 1b a2 d7 W34 = W33 ⊕ W30 = cb c3 aa 72 W35 = W34 ⊕ W31 = 3c be 0b 38	RotWord(W35) = be 0b 38 3c SubWord(X9) = ae 2b 07 eb Rcon (9) = 1B 00 00 00 y xor Rcon = b5 2b 07 eb
W36 = W32 ⊕ z = fd 0d 42 cb W37 = W36 ⊕ W33 = e1 6e 01 c W38 = W37 ⊕ W34 = c5 d5 4a 6e W39 = W38 ⊕ W35 = f9 6b 41 56	RotWord(W39) = 6b 41 56 f9 SubWord(X10) = 7f 83 b1 99 Rcon (10) = 36 00 00 00 y xor Rcon = 49 83 b1 99
W40 = W36 ⊕ z = b4 8e f3 52 W41 = W40 ⊕ W37 = ba 98 13 4e W42 = W41 ⊕ W38 = 7f 4d 59 20 W43 = W42 ⊕ W39 = 86 26 18 76	Empty

AES:

Start of Round	After SubBytes	After Shift Rows	After MixColumns	RoundKey
2200456789abcdeffedcba9876543210				22 00 45 67 89 ab cd ef fe dc ba 98 76 54 32 10
2c 00 34 ae ce 72 25 b6 e2 6b 17 4e d9 2b 55 88	71 63 18 e4 8b 40 3f 4e 98 7f f0 2f 35 f1 fc c4	71 40 f0 c4 8b 7f fc e4 98 f1 18 4e 35 63 3f 2f	16 3e 9d b0 94 8e 20 d6 75 07 8b c6 df 9d 59 5d	16 3e 9d b0 94 8e 20 d6 75 07 8b c6 df 9d 59 5d
cb bb aa 00 0e d2 ff 3f f3 ec f9 f9 f6 09 4c fa	1f ea ac 63 ab b5 16 75 0d ce 99 99 42 01 29 2d	1f b5 99 2d ab ce 29 63 0d 01 ac 75 42 ea 16 99	4e f3 f4 57 4e 34 92 c7 c0 95 d0 50 2e 2e 34 13	4e f3 f4 57 4e 34 92 c7 c0 95 d0 50 2e 2e 34 13
b3 2f 9f 42 29 b4 26 3b 21 fe 16 93 e6 d1 e7 77	6d 15 db 2c a5 8d f7 e2 fd bb 47 dc 8e 3e 94 f5	6d 8d 47 f5 a5 bb 94 2c fd 3e db e2 8e 15 f7 dc	e4 50 6a 8c 3f 43 59 83 9a 15 53 26 13 7a 11 c8	e4 50 6a 8c 3f 43 59 83 9a 15 53 26 13 7a 11 c8
0b ea 42 71 b7 79 c5 82 f3 44 09 e4 b2 d4 98 6e	2b 87 2c a3 a9 b6 a6 13 0d 1b 01 69 37 48 46 9f	2b b6 01 9f a9 1b 46 a3 0d 48 2c 13 37 87 a6 69	09 c0 25 ef 81 f6 c0 e0 fd fa 28 55 33 ba 5c aa	09 c0 25 ef 81 f6 c0 e0 fd fa 28 55 33 ba 5c aa
0a dd 29 20 0a d1 50 2e 1f 8c e2 59 70 62 1f 00	67 c1 a5 b7 67 3e 53 31 c0 64 98 cb 51 aa c0 63	67 3e 98 63 67 64 c0 b7 c0 aa a5 31 51 c1 53 cb	77 cb d7 c9 15 43 5a 78 ea 4a 68 36 62 f6 70 ec	77 cb d7 c9 15 43 5a 78 ea 4a 68 36 62 f6 70 ec
05 cc 77 1c ec 63 6a 63 f1 1c 92 21 3a 78 c9 51	6b 4b f5 9c ce fb 02 fb a1 9c 4f fd 80 bc dd d1	6b fb 4f d1 ce 9c dd 9c a1 bc f5 fb 80 4b 02 fd	5e 86 66 b0 79 0d 4c 2b 88 3d fa 5c 39 ed d3 33	5e 86 66 b0 79 0d 4c 2b 88 3d fa 5c 39 ed d3 33
15 d7 bc 0f cb 7c a6 8f 21 1a ea ef c8 44 7a 3d	59 0e 65 76 1f 10 24 73 fd a2 87 df e8 1b da 27	59 10 87 27 1f a2 da 76 fd 1b 65 73 e8 0e 24 df	22 cc 35 32 6f 43 88 b5 da 17 b9 84 22 47 d4 ac	22 cc 35 32 6f 43 88 b5 da 17 b9 84 22 47 d4 ac

Start of Round	After SubBytes	After Shift Rows	After MixColumns	RoundKey
fa 4e 44 2c 05 b0 13 0f 19 c3 32 8d 10 3a f6 ab	2d 2f 1b 71 6b e7 7d 76 d4 2e 23 5d ca 80 42 62	2d e7 23 62 6b 2e 42 71 d4 80 1b 76 ca 2f 7d 5d	29 ff 2a 77 97 80 52 33 45 94 f8 10 de 4e f8 ad	29 ff 2a 77 97 80 52 33 45 94 f8 10 de 4e f8 ad
8e ee 9e 4a 5a 62 7d b4 4b a2 5c 9e e2 05 7e 24	19 28 0b d6 be aa ff 8d b3 3a 4a 0b 98 6b f3 36	19 aa 4a 36 be 3a f3 d6 b3 6b 0b 8d 98 28 ff 0b	ab be 7d a7 0c 12 18 a7 46 f5 42 af a7 d9 48 72	ab be 7d a7 0c 12 18 a7 46 f5 42 af a7 d9 48 72
a4 eb 6e 71 ce a5 24 f6 8a 74 da 70 57 13 56 24	49 e9 9f a3 8b 06 36 42 7e 92 57 51 5b 7d b1 36	49 06 57 36 8b 92 b1 a3 7e 7d 9f 42 5b e9 36 51		49 06 57 36 8b 92 b1 a3 7e 7d 9f 42 5b e9 36 51

d.

If a small change in the key or plaintext is made, then algorithms like AES would be able to use the Avalanche effect. These small changes, even by a single bit, will drastically change the output of the ciphertext and with each round the difference increases.

e.

Round		Number of Bits
	1200456789abcdeffedcba9876543210 1300456789abcdeffedcba9876543210	1
0	1200456789abcdeffedcba9876543210 1300456789abcdeffedcba9876543210	1
1	1d0034aece7225b6e26b174ed92b5588 1c0034aece7225b6e26b174ed92b5588	1
2	7b6e7f640fd2ff3ff2ecf9f9f7094cfa 0b56472c0fd2ff3ff2ecf9f9f7094cfa	11
3	ce11a191a8499b964df86eeb2d11b8ed da1bab8f9a7bcdf23f6e8a995f4d96c3	55
4	64a8e7e4384ffa688b6b51e66abc7e8a 73037596bb7fb03b56d62d449d160439	69
5	bc8d55329e6fae08e6b5cb30b96a5044 a09a69fa2771d8452dc185c18e8f9ee2	69
6	94c5db6a304c88c61d25c50038516df6 9356c0601f2c407d79c00a326b2752bf	64
7	5b44e5f5c59baf88545ecb78a9cd3e84 df1804e12cd8fbd91f381fc6ed971056	58
8	1d3b28c74ace8076422143b967af6c68 e6e276587831da0f1c27c865cfcca829	71
9	e631cbf6f001f5fbd16e7ac78be99a9e 746e69ed49087ea5bee032ae9abced59	65
10	f6bff8c82a8f297d294f9782539562a7 23eb1ac227ff18027011b0a69dbdf33a	60

Round		Number of Bits
	1200456789abcdeffedcba9876543210 1200456789abcdeffedcba9876543310	1
0	1200456789abcdeffedcba9876543210 1200456789abcdeffedcba9876543310	1

Round		Number of Bits
1	1d0034aece7225b6e26b174ed92b5588 1d0034aece7225b6e26b174ed92b5488	1
2	7b6e7f640fd2ff3ff2ecf9f9f7094cfa 7b6e7f64d3ad5ce3f2ecf9f9f7094cfa	21
3	ce11a191a8499b964df86eeb2d11b8ed ae5181b188598ba6299cc22371f500b1	39
4	64a8e7e4384ffa688b6b51e66abc7e8a 9674ab3ec834a6602113c8058fd54ef8	65
5	bc8d55329e6fae08e6b5cb30b96a5044 b80929875a209651b42d3eca4745575f	65
6	94c5db6a304c88c61d25c50038516df6 decaeb706acf7c6f17f420e92ff3bf3f	59
7	5b44e5f5c59baf88545ecb78a9cd3e84 fd2202683589b490cf45c730497aca4e	62
8	1d3b28c74ace8076422143b967af6c68 a02bcffa1560ef8bae083ec663f32c5c	72
9	e631cbf6f001f5fbd16e7ac78be99a9e 242f8ed77387d3507d92ae44b265eec2	58
10	f6bff8c82a8f297d294f9782539562a7 30bd25129bd015a01f7eed377766dd4a	74

Problem 3

a.
Parallel encryption is not possible since every encryption requires a previous cipher. In decryption all blocks can be processed in parallel.
b.
All ciphertext blocks from P_2 and on will be affected .
c.
The error would effect all ciphertext blocks. This will cause the reciever to recieve a completely different result than what is expected.
d.
It is possible to perform encryption operations in parallel on multiple blocks of plaintext in CTR mode for both Encryption and Decryption.
e.
No, because Counter mode does not use chaining no other blocks would be affected if an error in a block of transmitted ciphertext occurs.

f.
The only ciphertext block that would be affected is the one which the error initially propagated. Beyond that block no other blocks would be affected. The reciever would be given an incorrect value for only the ciphertext block that was affected all other outputs should be correct.

Problem 4

a. We know that $P(0) = 0.5 - \vartheta$ and $P(1) = 0.5 + \vartheta$, thus we can derive:

Pair	Probability
00	$(0.5 - \vartheta)^2 = 0.25 - \vartheta + \vartheta^2$
01	$(0.5 - \vartheta) \times (0.5 + \vartheta) = 0.25 - \vartheta^2$
10	$(0.5 + \vartheta) \times (0.5 - \vartheta) = 0.25 - \vartheta^2$
11	$(0.5 + \vartheta)^2 = 0.25 - \vartheta + \vartheta^2$

b.

Because the pairs 01 and 10 have equal probability within the initial sequence, the modified sequence will convert these to 0 and 1, thus P'(0) = P'(1)=0.5

c.

The probability of a pair being discarded is equal to the probability that a pair is either 00 or 11, so $P(00) + P(11) = (0.25 - \vartheta + \vartheta^2) + (0.25 + \vartheta + \vartheta^2) = 0.5 + 2\vartheta^2$.

Thus, the expected number of bits to produce x output bits is as follows:

$2x/(0.5 - 2\vartheta^2)$

$= x/(.25 - \vartheta^2)$ input bits

d.

By making the modified sequence consider overlapping pairs of bits, the output bit stream will be a sequence of alternating 1's and 0's which is completely predictable.

e.

For the sequence of input bits $a_1, a_2, ..., a_n$ the output bit is defined as $\oplus$

$$b = a_1 \oplus a_2 \oplus ... \oplus a_n$$

f.

if each group consists of 2 bits, then the probability of an output being 1 will now be:

$$0.5 - 2\vartheta^2$$

g.

If each group consists of 4 bits, then the probability of an output being 1 will now be:

$$0.5 - 8\vartheta^4$$

Problem 5

a.

Fermat's Theorem is if *p* is prime and a is a positive integer not divisible by *p* then:

$$a^{p-1} \equiv 1 (\mod p)$$

If we consider the set of positive integers less than

p : $\{1, 2, ..., p-1\}$

We then multiply each element in the set by a modulo p to get set

$$X = \{a \mod p, 2a \mod p, ..., (p-1)a \mod p\}$$

If we assume that $ja \equiv ka(\mod p)$ where $1 \leq j < k \leq p-1$ then we determine that None of the elements of X is equal to zero because p does not divide a and no two integers in X are equal.

Because a is relatively prime to p, we can eliminate both sides of the equation:

$$\begin{cases} a = q_1 b + r_1 & 0 < r_1 < b \\ b = q_2 r_1 + r_2 & 0 < r_2 < r_1 \\ r_1 = q_3 r_2 + r_3 & 0 < r_3 < r_2 \\ \vdots & \vdots \\ r_{n-2} = q_n r_{n-1} + r_n & 0 < r_n < r_{n-1} \\ r_{n-1} = q_{n+1} r_n + 0 & \\ d = gcd(a, b) = r_n & \end{cases} \tag{1}$$

which results in $j \equiv k(\mod p)$. This last equality is impossible, because j and k are both positive integers less than p.

$$\therefore$$

We know that the $(p-1)$ elements of X are all positive integers with no two elements equal. Thus the X consists of the set of integers $1, 2, ..., p-1$ in some order. Multiplying the numbers in both sets (p and X) and taking the result mod p yields.

$$\begin{gathered} a \times 2a \times \cdots \times (p-1)a \equiv [(1 \times 2 \times \cdots \times (p-1))](\mod p) \\ a^{p-1}(p-1)! \equiv (p-1)!(\mod p) \\ a^{p-1} \equiv 1(\mod p) \end{gathered}$$

b.

Euler's theorem states that for every a and n that are relatively prime:

$$a^{\phi(n)} \equiv 1(\mod n)$$

the above equation is true if n is prime, because in that case, $\phi(n) = (n-1)$ and Fermat's theorem holds. It will also hold for any integer n. Recall that $\phi(n)$ is the number of positive integers less than n that are relatively prime to n. If we consider a set of R integers:

$$R = \{x_1, x_2, ..., x_{\phi(n)}\}$$

That is, each element x_i of R is a unique positive integer less than n with $\gcd(x_i, n) = 1$. We then multiply each element by $a \mod n$:

$$S = \{(ax_1 \mod n), (ax_2 \mod n_), ..., (ax_{\phi(n)} \mod n)\}$$

The set S is a permutation of R, by the following line of reasoning

1. Because a is relatively prime to n and x_i is prime to n, ax_1 must also be relatively prime to n. Thus, all the members of S are integers that are less than n and that are relatively prime to n.
2. There are no duplicates in S. If we examine equation 2.5 $if (a \times b) \equiv (a \times c)(\mod n) then\, b \equiv c(\mod n) if\; a$ is relatively prime to *n*. If $ax_i \mod n = ax_j \mod n, then\; x_i = x_j$

$$\therefore$$

$$\begin{gathered} \prod_{i=1}^{\phi(n)} (ax_i \mod n) = \prod_{i=1}^{\phi(n)} x_i \\ \prod_{i=1}^{\phi(n)} ax_i \equiv \prod_{i=1}^{\phi(n)} x_i(\mod n) \\ a^{\phi(n)} \times [\prod_{i=1}^{\phi(n)} x_i] \equiv x_i(\mod n) \\ a^{\phi(n)} \equiv 1(\mod n) \end{gathered}$$

c.

Suppose $n > 2$. If n has an odd prime factor, say p; then

$$n = p^k m, (m, p) = 1$$

and

$$\varphi(n) = \varphi(p^k)\varphi(m) = (p-1)p^{k-1}\varphi(m)$$

with $p-1$ even. If n has no odd prime factors then $n = 2^k$ with $k > 1$ so $\varphi(2^k) = 2^{k-1}$ is even.
In short
if $\gcd(k, n) = 1$, then $\gcd(n-k, n) = 1$ as well, so (for n>2) all the numbers relatively prime to n can be matched up into pairs $\{k, n-k\}$. So $\phi(n)$ is even.

Problem 6

a.

a	a^2	a^3	a^4	a^5	a^6	a^7	a^8	a^9	a^10	a^11	a^12
1	1	1	1	1	1	1	1	1	1	1	1
2	4	8	3	6	12	11	9	5	10	7	1

a	a^2	a^3	a^4	a^5	a^6	a^7	a^8	a^9	a^10	a^11	a^12
3	9	1	3	9	1	3	9	1	3	9	1
4	3	12	9	10	1	4	3	12	9	10	1
5	12	8	1	5	12	8	1	5	12	8	1
6	10	8	9	2	12	7	3	5	4	11	1
7	10	5	9	11	12	6	3	8	4	2	1
8	12	5	1	8	12	5	1	8	12	5	1
9	3	1	9	3	1	9	3	1	9	3	1
10	9	12	3	4	1	10	9	12	3	4	1
11	4	5	3	7	12	2	9	8	10	6	1
12	1	12	1	12	1	12	1	12	1	12	1

b.

Discrete	Logarithms	to	the	base	2	Modulo	29													
a	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	2
$\log_{2,29}(a)$	0	1	5	2	22	6	12	3	10	23	25	7	18	13	27	4	21	11	9	2