# Fall 2023 CIS 3362 Final Exam (12/6/2023) Solutions

1) (5 pts) The ciphertext **NCJAE** was encrypted via the Shift Cipher with a key of 11. What is the corresponding plaintext?

N – 11 = 13 – 11 = 2 (C)
C – 11 = 2 – 11 = -9 = 17 (mod 26) (R)
J – 11 = 9 – 11 = -2 = 24 (mod 26) (Y)
A – 11 = 0 – 11 = -11 = 15 (mod 26) (P)
E – 11 = 4 – 11 = -7 = 19 (mod 26) (T)

**CRYPT (Grading: 1 pt per letter)**

2) (5 pts) Encrypt the plaintext **ROCKETLAUNCHNOW** using the Vigenere Cipher with a keyword of **SIGHT.**

```
R  O  C  K  E  T  L  A  U  N  C  H  N  O  W
S  I  G  H  T  S  I  G  H  T  S  I  G  H  T

17 14  2 10  4 19 11  0 20 13  2  7 13 14 22
18  8  6  7 19 18  8  6  7 19 18  8  6  7 19
-----------------------------------------
35 22  8 17 23 37 19  6 27 32 20 15 19 21 41 (under mod →)

 9 22  8 17 23 11 19  6  1  6 20 15 19 21 15
 J  W  I  R  X  L  T  G  B  G  U  P  T  V  P
```

**JWIRXLTGBGUPTVP (Grading : floor(correct/3) , so have to get all to get 5)**

3) (15 pts) Recall that in Radix-64, 'A' – 'Z' are encoded as 0 to 25, 'a' to 'z' are encoded as 26 to 51, '0' to '9' are encoded as 52 to 61, '+' is encoded as 62 and '/' is encoded as 63. The ciphertext **`FjOOvDCbHlaLdLi`** was encrypted via the Affine Cipher with the **encryption keys of** a = 23 and b = 49. (Thus, the encryption function was f(x) = (23x + 49)%64.) First, find the corresponding decryption function (10 pts), then use this decrypt the given ciphertext to recover the plaintext (5 pts). **Note: Due to the space allowed to work on the problem, show ALL of your work for the first part, but just do the second part on your calculator completely and the grade will be based on the answer only. (Also, characters 3 and 4 are uppercase letter O's, and character 10 is a lowercase letter L.)**

First, let's figure out the decryption function:

$x = (23y + 49) \bmod 64$
$(x - 49) = 23y \bmod 64$
$y = 23^{-1}(x - 49) \bmod 64$, thus we must determine $23^{-1} \bmod 64$ via the Extended Euclidean Algorithm.

$64 = 2 \times 23 + 18$
$23 = 1 \times 18 + 5$
$18 = 3 \times 5 + 3$
$5 = 1 \times 3 + 2$
$3 = 1 \times 2 + 1$

$3 - 2 = 1$
$3 - (5 - 3) = 1$
$2 \times 3 - 1 \times 5 = 1$ (2 steps at once)
$2(18 - 3 \times 5) - 1 \times 5 = 1$
$2 \times 18 - 7 \times 5 = 1$ (2 steps at once)
$2 \times 18 - 7(23 - 18) = 1$
$9 \times 18 - 7 \times 23 = 1$ (2 steps at once)
$9(64 - 2 \times 23) - 7 \times 23 = 1$
$9 \times 64 - 25 \times 23 = 1$ (2 steps at once)

Take this last equation mod 64 to get $-25 \times 23 \equiv 1 \pmod{64}$. It follows, that

$$23^{-1} \equiv -25 \equiv 39 \ (mod\ 64)$$

Thus, the decryption function is $f^{-1}(x) = 39(x - 49) = 39x - 1911 \equiv 39x + 9 \pmod{64}$
Next, let's convert the ciphertext to numbers:

5, 35, 14, 14, 47, 3, 2, 27, 7, 37, 26, 11, 29, 11, 34

As mentioned above, instead of writing out each formula, directly type into a calculator. One is shown here for convenience: $f^{-1}(5) = 39(5) + 9 = 195 + 9 = 204 = 12 \bmod 64$ (M)

Here are all 15 plaintext numbers:

12, 30, 43, 43, 50, 62, 23, 38, 26, 44, 63, 54, 52, 54, 55 converting via Radix-64 codes we get:

**Merry+Xmas/2023**

**Grading: 2 pts inverse set up, 3 pts Euclid, 5 pts EEA and any valid decryption function (no need to map to 0-63), 5 pts for message (do floor(correct/3) again.**

4) (6 pts) Consider the following IP matrix for a DES-like cipher with a block size of 16 bits:

$$IP = \begin{bmatrix} 14 & 11 & 5 & 4 \\ 13 & 15 & 10 & 2 \\ 6 & 3 & 16 & 1 \\ 7 & 9 & 12 & 8 \end{bmatrix}$$

What is the minimum number of times that IP has to be applied to an arbitrary input, x, to guarantee that the result is x? (Note: This question has nothing to do with DES. **Also, don't try to keep on applying the matrix to an input to get the answer.**)

First, let's get a mapping for where the bits are moving:

14→1, 11→2, 5→3, 4→4, 13→5, 15→6, 10→7, 2→8, 6→9, 3→10, 16→11, 1→12, 7→13, 9→14, 12→15, 8→16. (First number is old bit position, second number is new bit position.)

Now, let's represent this data in chains, just following the arrows:

14→1→12→15→6→9→14 (chain length is 6)
11→2→8→16→11 (chain length is 4)
5→3→10→7→13→5 (chain length is 5)
4→4 (chain length is 1)

Thus, some bits return to their original location every 6 applications of IP, others every 4 applications of IP, others every 5 applications of IP, or the 4th bit, which stays put! This means that all 16 bits are guaranteed to return to their original locations in lcm(6, 4, 5) = 60 applications of IP. (You can work out the least common multiple by noting that 12 is the lcm of 6 and 4, and then 12 and 5 are relatively prime, so we can just multiply those.)

**60, Grading: 2 pts writing out mappings, 3 pts stating correct cycle lengths, 1 pt final answer**

5) (8 pts) Let the input to the S-boxes in DES be 3A7 DC9 61E F4B, in hex. What is the output produced by the S-boxes. Express your result in hex.

$S_1(001110) = S_1[0][7] = 8$
$S_2(100111) = S_2[3][3] = 1$
$S_3(110111) = S_3[3][11] = 3$
$S_4(001001) = S_4[1][4] = 6$
$S_5(011000) = S_5[0][12] = 13(D)$
$S_6(011110) = S_6[0][15] = 11(B)$
$S_7(111101) = S_7[3][14] = 3$
$S_8(001011) = S_8[1][5] = 3$

**8136DB33 (Grading: 1 pt per hex character, only award if there is reasonable evidence that the correct row and column were deciphered, so that copied answers don't get points.)**

6) (10 pts) Let the input to the MixCols step of AES be the state matrix shown below. Determine the entry in row 4 column 2 **right after** the MixCols step has been completed. **Express your answer in HEX.**

| | | | |
|---|---|---|---|
| 51 | F9 | 41 | C6 |
| 77 | 80 | 26 | E6 |
| 08 | E0 | 4D | 1B |
| 0C | 6E | DE | 3F |

The sum of products we have to calculate is 03 x F9 + 01 x 80 + 01 x E0 + 02 x 6E

03 x F9 = (02 x F9) + (01 x F9)
01 x 80 = 80
01 x E0 = E0
02 x 6E = 0110 11100 = 1101 1100 (DC)

02 x F9 = 1111 1001 0 = Thus, 03 x F9 = E9 + F9 = 10

```
    1111 0010
         1 1011
    ------------
    1110 1001 (E9)
```

Thus, our final sum is 10 + 80 + E0 + DC = 90 + 3C = **AC**

**Grading:**

**3 pts for writing correct sum of products (give automatic 2 out of 10 if this is incorrect)**

**2 pts to evaluate 02 x 6E correctly**

**3 pts to evaluate 03 x F9 correctly**

**2 pts for final answer (if it's not in HEX take off 1 of these 2 pts)**

7) (10 pts) Let a and n be arbitrary positive integers. Create a set S containing the remainders when each of the values a, 2a, 3a, 4a, 5a, …, na is each divided by n. **With proof**, determine the number of values in the set S. (The result is an expression in terms of a and n.) For example, when a = 2 and n = 6, the answer is 3, since the three remainders in S for this case are 0, 2 and 4. (Note: 8 pts for the proof, 2 pts for the answer.)

S will contain $\frac{n}{\gcd(a,n)}$ values. (Trying out some examples, such as n = 6, a = 4 will help establish a reason to make this guess.)

By definition of gcd, gcd(a, n) | a.
By definition of integer multiplication, it follows that gcd(a, n) | ai, for any integer i.

Thus, each value in S **MUST BE** a multiple of gcd(a, n). The number of these non-negative multiples that are less than n is exactly $\frac{n}{\gcd(a,n)}$. (Let d = gcd(a, n), then these multiples are 0 x d, 1 x d, 2 x d, 3 x d, …, $\left(\frac{n}{\gcd(a,n)} - 1\right) \times d = n - d$. This establishes that the **MAXIMUM POSSIBLE SIZE** of S is $\frac{n}{\gcd(a,n)}$.

Before we continue, let $a' = \frac{a}{d}$ and $n' = \frac{n}{d}$, be integers, noting that gcd(a', n') = 1.

The second part of the proof is showing that each multiple of d previously described is a remainder when one of the numbers in the original list is divided by n. Thus, we must prove, for any value di, where, where $0 \leq i < \frac{n}{\gcd(a,n)}$, there exists some integer k, where $1 \leq k \leq n$, such that

$$ak \equiv di \ (mod\ n)$$

Let i arbitrarily chosen above. This equation is equivalent to proving that $n \mid (ak - di)$, for some k in the given range. Notice that there is a common factor in the two expressions n and ak – di. This common factor is d. If we divide d out of each, it remains to be proven that there exists a k such that $n' | (a'k - i)$. Turning this back into a mod equation, equivalently, we must prove that there exists a k such that

$$a'k \equiv i (mod\ n')$$

Since gcd(a', n') = 1, by the Extended Euclidean Algorithm, we know that a modular inverse for a' exists and we can solve for k as follows: $k \equiv \left(a'^{-1}\ mod\ n'\right) i\ (mod\ n')$. There is a guarantee that there is at least one value of k in between 0 and n' that satisfies this equation, which also means that there is at least one value in the set 1 through n, inclusive that satisfies this equation. (Note that if k = 0 satisfies the equation, so does k = n.)

It follows that the size of the set S is precisely $\frac{n}{\gcd(a,n)}$. **Grading: 2 pts for result, 4 pts for proving the set can't be bigger than this, 4 pts for proving that the set is actually this size.**

8) (12 pts) In an RSA system, n = 403 and e = 133. What is d?

Via trial and error, 403 = 13 x 31.
Thus $\phi(n) = \phi(13 \times 31) = \phi(13) \times \phi(31) = 12 \times 30 = 360.$

$d = 133^{-1} \bmod 360$

360 = 2 x 133 + 94
133 = 1 x 94 + 39
94 = 2 x 39 + 16
39 = 2 x 16 + 7
16 = 2 x 7 + 2
7 = 3 x 2 + 1

7 – 3 x 2 = 1
7 – 3(16 – 2 x7) = 1
7 – 3 x 16 + 6 x 7 = 1
7 x 7 – 3 x 16 = 1
7(39 – 2 x16) – 3 x 16 = 1
7 x 39 – 14 x 16 – 3 x 16 = 1
7 x 39 – 17 x 16 = 1
7 x 39 – 17(94 – 2 x 39) = 1
7 x 39 – 17 x 94 + 34 x 39 = 1
41 x 39 – 17 x 94 = 1
41(133 – 94) – 17 x 94 = 1
41 x 133 – 41 x 94 – 17 x 94 = 1
41 x 133 – 58 x 94 = 1
41 x 133 – 58(360 – 2 x 133) = 1
41 x 133 – 58 x 360 + 116 x 133 = 1
157 x 133 – 58 x 360 = 1

Take this equation mod 360 to yield

$157 \times 133 \equiv 1 \pmod{360}$

It follows that **d = 157.**

**Grading: 1 pt factor 403, 2 pts phi(403), 3 pts Euclidean, 5 pts Extended, 1 pt final answer**

9) (5 pts) The quantum cryptography method described in class allows Alice and Bob to exchange a secret key while verifying that no one else was listening to the exchange. Alice and Bob have an idea to use a similar method to exchange an actual message. Since Bob's reader will be correct 50% of the time, Alice will send 3n bits, where n is the number of bits that Alice wants to communicate to Bob. She will send the bits in groups of three, sending the same bit three times. For example, if she wants to send the message 101, she'll actually send 111000111. The first of these bits will be sampled to detect eavesdroppers. The second and third bits will be so that Bob gets one copy of the intended message (since he chooses the right reader 50% of the time). Alice and Bob will both choose randomly selected readers for sending and receiving each bit. What are some flaws with this system?

(1) Recall that sampled bits are discussed over an insecure line. But the way this is described, the sampled bits ARE the message, so anyone listening on that insecure line could get the message. (In retrospect, I wish I had said that these bits were random and that the message bits were repeated twice…)

(2) If there is a pattern of sampled bits, Eve might be able to simply "not listen" for exactly the sampled bits (every third one) and go undetected but still listen in on the bits that matter.

(3) More importantly though, Just because Bob guesses right 50% on average doesn't mean that he'll guess the same reader as Alice at least once out of every two times. In fact, due to probability, there is a 25% chance he'll use the wrong reader both times. In cases where he uses the wrong reader once, half the time he receives 1 right bit and 1 wrong bit, with know way of knowing which bit was the intended one. In cases where he uses the wrong reader twice, all three options are possible (getting 2 copies of the right bit, 1 of each bit, 2 copies of the wrong bit). There's no way for Bob to know which of these cases he is in when he receives the bits. If he gets 2 copies of the same bit, it's more likely that bit was the correct one, but if he gets two different bits, there is no way for him to know which bit was sent by Alice.

**Grading: full credit for any of these three reasons clearly explained, partial as needed.**

10) (8 pts) Let $p_1$, $p_2$ and $p_3$ be the three most commonly used passwords. Assume that $p_1$ is used by 5% of all logins, $p_2$ is used by 2% of all logins and $p_3$ is used by 1% of all logins. Let there be 30 users on a system. (a) What is the probability that none of the users has used $p_1$, $p_2$ or $p_3$ as their password? (b) What is the probability that at least two people are using $p_1$ as their password? Please give both answers as decimals in between 0 and 1, rounded to 6 digits. In your work, show what you typed into your calculator.

(a) The probability a single user doesn't use any of the three passwords is 1 - .05 - .02 - .01 = .92. It follows that the probability none of the users has used any of these is $.92^{30} \sim 0.081966$.

(b) We can more easily calculate the probability that 0 or 1 people in the group are using $p_1$. Using the logic from before, the probability that 0 people are using $p_1$ is $.95^{30}$. The probability that exactly 1 person is using $p_1$ is $30 \times .95^{29} \times .05$. The reasoning here is that there are 30 ways in which one person can use $p_1$ – we can choose WHICH user out of 30 has this password. Once we do that the other 29 are locked into having something else. The probability of each of these 30 possibilities is simply the product of that one person choosing $p_1$ (which is .05) with the probability that the other 29 don't choose it, which is $.95^{29}$. Thus, the probability that either 0 or 1 users have $p_1$ as their password is $.95^{30} + 30 \times .05 \times .95^{29} \sim 0.553542$. We want to answer the opposite question, thus, the desired answer is 1 - 0.553542 = 0.446458.

Answer to (a) : **0.081966** Answer to (b): **0.446458**

**Grading: 1 pt for theoretical expression for (a), 1 pt for plugging in calculator and presenting to 6 decimal places, 1 pt for no one p1, 3 pts for one person p1, 1 pt to subtract from 1, 1 pt for answer to 6 decimal places**

11) (10 pts) Let P = (19, 11) and Q = (33, 32) on the Elliptic Curve $E_{37}(4, 31)$. What is the result of P + Q?

$$\lambda = \frac{32 - 11}{33 - 19} = \frac{21}{14} = \frac{3}{2} = 3 \times (2^{-1} mod 37)$$

Run the Extended Euclidean Algorithm (but this one's easy):

37 = 18 x 2 + 1
37 – 18 x 2 = 1, take this equation mod 37
-18 x 2 ≡ 1 (mod 37)
It follows that $2^{-1} \equiv -18 \equiv 19\ (mod\ 37)$

Thus, $\lambda = 3 \times 19 = 57 \equiv 20\ (mod\ 37)$

$$x_R = 20^2 - 19 - 33 = 400 - 52 = 348 \equiv 15\ (mod\ 37)$$
$$y_R = 20(19 - 15) - 11 = 80 - 11 = 69 \equiv 32\ (mod\ 37)$$

It follows that the result of the sum is the point **(15, 32).**

**Grading: 2 pts plug in lambda**
**2 pts simplify lambda to get 20 via EEA or other means**
**3 pts plug in and get x**
**3 pts plug in and get y**

12) (5 pts) Who was the original inventor of both RSA Encryption and the Diffie-Hellman key exchange? (If you don't remember the name, explain who he worked for and the circumstances under which he invented these protocols.) Why did this person not get the notoriety that Rivest, Shamir, Adleman, Diffie and Hellman got?

Clifford Cocks – he worked for the British Government, more specifically, the Government Communications Headquarters (though I don't think I mentioned this name in class), the U.K. equivalent to the U. S. NSA. Because he was working for the government, his work was classified by the British government and he wasn't allowed to share it with anyone outside of GCHQ. Because of this, he wasn't allowed to capitalize on his invention publicly and therefore, didn't gain notoriety for it.

**Grading:**

**3 pts for name, 2 pts if no name but remembered it was someone working for the British**

**2 pts for explaining that he wasn't allowed to share the information publicly.**

13) (1 pt) UCF's Football team will play in the Gasparilla Bowl. The last time UCF played in this bowl game was 2021 when they defeated the University of Florida. What Bowl was UCF invited to in 2021?

**Gasparilla** (Grading: Give to all)