CESI_
ÉCOLE D'INGÉNIEURS

CER 6

The Mystery of the Scottish Castle

Part 5

October 1, 24

Cyusa Adnan Junior

1. Roles

- ***Facilitator***: Andrew
- ***Resource and Time Manager***: Maxime
- ***Scribe***: Palal
- ***Secretary***: Junior

2. Keywords

- Encrypted message : A message that has been transformed into a coded form to prevent unauthorized access.
- ASCII : : A message that has been encrypted using the **ASCII** (American Standard Code for Information Interchange) values of characters.
- Shift : refers to a technique where the letters of the alphabet are shifted by a certain number of positions to create a cipher.
- Deciphering : The process of decoding or converting an encrypted or encoded message back into its original, understandable form, often by reversing the encryption process.
- Artifacts : refer to unintended or residual effects or distortions in data, messages, or images due to encryption or other processing steps.
- Cryptic Message : A message that is intentionally obscure or difficult to understand, often using codes, symbols, or a form of secret writing.

- Latitude and Longitude are geographical coordinates used to pinpoint a location on Earth's surface.
- Mystical weight : refer to the **metaphorical or symbolic significance** that certain objects, numbers, or concepts hold in mystical, esoteric, or spiritual traditions.

3. Context

We received a letter from Alexander asking for our help to decipher encrypted messages to find treasures that have been hidden by his ancestors.

4. Issue

How can we come up with a standardized python code that can decipher ancient scrolls taking into consideration the hints that have been provided by Alexander MacLeod?

5. Constraints

- The crypted messages are largely numerical.
- The problem statement limits us to using python code to decrypt the messages on the scrolls.
- Limited scrolls of numbers to work with to come up with a generalization.

6. Generalization

- Arithmetic of Cryptography

7. Deliverable

- Python script
- Written report (CER)

8. Hypothesis

- N and E might prove to be tricky in aiding us to decode the numbers.
- Data might be missing due to old scrolls.
- Deciphering will involve Bitwise operations.
- A single script can be used to decipher both sets of decrypted data
- The script we come up with not work across multiple scroll
- The data will be impossible to decipher
- The first scroll requires Ceasar encryption
- We would make use of a Euclidean algorithm

9. Action Plan

1. Define keywords
2. Analyse resources
3. Compare various encryption algorithms to find the most suitable one for the context at hand
4. Theoretically derive, using mathematical methods, a decryption solution
5. Implement code to get coordinates from parchment based on mathematical derivation.
6. Validate hypothesis
7. Conclusion

Resource Analysis

ASCII

In ASCII, the letter 'A' is represented by 65. An ASCII-encrypted message might convert "HELLO" to "72 69 76 76 79" before applying further encryption.

⊞	65	101	41	01000001	A	&#65;	
⊞	66	102	42	01000010	B	&#66;	
⊞	67	103	43	01000011	C	&#67;	
⊞	68	104	44	01000100	D	&#68;	
⊞	69	105	45	01000101	E	&#69;	
⊞	70	106	46	01000110	F	&#70;	
⊞	71	107	47	01000111	G	&#71;	
⊞	72	110	48	01001000	H	&#72;	
⊞	73	111	49	01001001	I	&#73;	
⊞	74	112	4A	01001010	J	&#74;	
⊞	75	113	4B	01001011	K	&#75;	
⊞	76	114	4C	01001100	L	&#76;	
⊞	77	115	4D	01001101	M	&#77;	
⊞	78	116	4E	01001110	N	&#78;	
⊞	79	117	4F	01001111	O	&#79;	
⊞	80	120	50	01010000	P	&#80;	
⊞	81	121	51	01010001	Q	&#81;	
⊞	82	122	52	01010010	R	&#82;	
⊞	83	123	53	01010011	S	&#83;	
⊞	84	124	54	01010100	T	&#84;	
⊞	85	125	55	01010101	U	&#85;	
⊞	86	126	56	01010110	V	&#86;	
⊞	87	127	57	01010111	W	&#87;	
⊞	88	130	58	01011000	X	&#88;	
⊞	89	131	59	01011001	Y	&#89;	

Prime numbers

Euclid's Algorithm:

- **Purpose:** It is used to compute the greatest common divisor (GCD) of two integers.
- **Algorithm:**
    1. Take two numbers, say a and b, where a>ba > ba>b.
    2. Replace aaa with a%ba \% ba%b (the remainder of dividing a by b).
    3. Repeat the process with aaa and bbb until b=0b = 0b=0.
    4. When b=0b = 0b=0, aaa will be the GCD.

Bézout's Identity:

- **Definition:** If aaa and bbb are integers, and ddd is their GCD, then Bézout's identity states that there exist integers xxx and yyy such that: ax+by=dax + by = dax+by=d In other words, the GCD of aaa and bbb can be expressed as a linear combination of aaa and bbb.

Is Prime:

- **Definition:** A prime number is an integer greater than 1 that has no positive divisors other than 1 and itself.
- Algorithm to Check Prime:
    1. If n≤1n \leq 1n≤1, it is not prime.
    2. Check divisibility from 2 to n\sqrt{n}n. If nnn is divisible by any number in this range, then it is not prime.
    3. If no divisors are found, nnn is prime.

Mersenne Prime:

- Definition: A Mersenne prime is a prime number of the form Mn=2n−1M_n = 2^n - 1Mn=2n−1, where nnn is a positive integer.
- Example: If n=3n = 3n=3, M3=23−1=7M_3 = 2^3 - 1 = 7M3 =23−1=7, and 7 is a prime number, making it a Mersenne prime.

Arithmetic Theorems

1. **Fundamental Theorem of Arithmetic:**
   - **Statement**: Every integer greater than 1 is either a prime number or can be factored uniquely (up to the order of the factors) into prime numbers.
   - **Example**: 60=22×3×560 = 2^2 \times 3 \times 560=22×3×5. There is only one such prime factorization of 60.
2. **Euclid's Lemma:**
   - **Statement**: If a prime number ppp divides the product ababab, then ppp must divide at least one of aaa or bbb.
   - **Application**: This lemma is crucial in proving the uniqueness of the prime factorization in the Fundamental Theorem of Arithmetic.
3. **Chinese Remainder Theorem:**
   - **Statement**: If you have a system of simultaneous congruences: x≡a1(modn1)x \equiv a_1 \pmod{n_1}x≡a1(modn1) x≡a2(modn2)x \equiv a_2 \pmod{n_2}x≡a2(modn2) with

n1n_1n1 and n2n_2n2 coprime (GCD is 1), then there exists a unique solution modulo n1×n2n_1 \times n_2n1×n2.

- **Example**: Solve x≡2(mod3)x \equiv 2 \pmod{3}x≡2(mod3) and x≡3(mod5)x \equiv 3 \pmod{5}x≡3(mod5). The solution is x=8x = 8x=8 modulo 15.

4. **Fermat's Little Theorem:**
   - **Statement**: If ppp is a prime number and aaa is an integer not divisible by ppp, then: ap−1≡1(modp)a^{p-1} \equiv 1 \pmod{p}ap−1≡1(modp)
   - **Example**: For p=7p = 7p=7 and a=3a = 3a=3, Fermat's Little Theorem says that 36≡1(mod7)3^6 \equiv 1 \pmod{7}36≡1(mod7).
5. **Bezout's Theorem:**
   - **Statement**: If aaa and bbb are integers with greatest common divisor ddd, then there exist integers xxx and yyy such that ax+by=dax + by = dax+by=d.
   - **Application**: This theorem is useful for solving Diophantine equations (linear equations in two or more unknowns).
6. **Pythagorean Theorem (Special Case in Arithmetic):**
   - **Statement**: In a right-angled triangle, the square of the hypotenuse is equal to the sum of the squares of the other two sides.
   - **Arithmetic Application**: It's closely related to the study of Pythagorean triples, where integer solutions to the equation a2+b2=c2a^2 + b^2 = c^2a2+b2=c2 are sought.
7. **Dirichlet's Theorem on Primes in Arithmetic Progressions:**

- **Statement**: If a,a and d,d are coprime, then the arithmetic progression a,a+d,a+2d,a+3d,…a, a + d, a + 2d, a + 3d, \dotsa,a+d,a+2d,a+3d,… contains infinitely many prime numbers.
- **Example**: The sequence 5,11,17,23,29,…5, 11, 17, 23, 29, \dots5,11,17,23,29,… contains infinitely many primes since 5 and 6 (the common difference) are coprime.

8. **Euler's Totient Theorem:**

- **Statement**: If aaa and nnn are coprime, then: $a^{\phi(n)} \equiv 1 \pmod{n}$ where ϕ(n)\phi(n)ϕ(n) is Euler's totient function, which counts the number of integers up to n that are coprime with n.
- **Example**: If n=10n = 10n=10, ϕ(10)=4\phi(10) = 4ϕ(10)=4, and for a=3a = 3a=3, 34≡1(mod10)3^4 \equiv 1 \pmod{10}34≡1(mod10).

9. A **Sophie Germain prime** is a special kind of prime number named after the French mathematician **Sophie Germain**. A prime number ppp is called a Sophie Germain prime if **2p + 1** is also a prime number.

   Definition:

   - A prime number ppp is a **Sophie Germain prime** if ppp is prime and 2p+12p + 12p+1 is also prime.
     - For example, if p=11p = 11p=11, then 2×11+1=232 \times 11 + 1 = 232×11+1=23, and since 23 is also a prime number, 11 is a Sophie Germain prime.

Encryption Algorithms

Encryption and decryption algorithms are used to secure data by converting it into an unreadable format (encryption) and then back to its original form (decryption) using keys. Below are some common **encryption and decryption algorithms** along with brief descriptions of how they work:

1. Symmetric Key Encryption

- **Definition**: Symmetric encryption uses the same key for both encryption and decryption.
- **Examples**:

a. AES (Advanced Encryption Standard):

- **Algorithm**: AES is a widely used symmetric encryption algorithm. It encrypts data in blocks of 128 bits using keys of 128, 192, or 256 bits. The data is scrambled through several rounds of substitution and permutation.
- **Strength**: Strong and efficient, used in various security protocols such as SSL/TLS.
- **Decryption**: The ciphertext is transformed back to plaintext using the same key through the inverse process of AES.

b. DES (Data Encryption Standard):

- **Algorithm**: DES encrypts data in 64-bit blocks using a 56-bit key. It operates on the data using 16 rounds of substitution and permutation.
- **Strength**: It is now considered insecure due to its short key length and can be cracked with modern computational power.

- **Decryption**: The decryption process is the same as encryption but uses the key in reverse.

c. 3DES (Triple DES):

- **Algorithm**: A stronger version of DES, it applies the DES algorithm three times using three different keys (or two keys).
- **Strength**: More secure than DES, but slower and largely replaced by AES.
- **Decryption**: The ciphertext is decrypted by applying DES in reverse order.

2. Asymmetric Key Encryption

- **Definition**: Asymmetric encryption uses two keys: a **public key** for encryption and a **private key** for decryption.
- **Examples**:

a. RSA (Rivest–Shamir–Adleman):

- **Algorithm**: RSA is based on the difficulty of factoring large prime numbers. The public key is used to encrypt the message, and the private key is used to decrypt it. The keys are generated from two large prime numbers.
- **Strength**: Strong and widely used in secure communications (like SSL certificates).
- **Decryption**: Decryption is done by using the private key to reverse the RSA mathematical operation.

b. ECC (Elliptic Curve Cryptography):

- **Algorithm**: ECC is based on the mathematics of elliptic curves over finite fields. It provides the same level of security as RSA with smaller key sizes, making it more efficient.
- **Strength**: Increasingly popular for secure communications due to its efficiency and lower computational load.
- **Decryption**: The private key, which is associated with the elliptic curve, is used to decrypt the message encrypted with the public key.

3. Hash Functions (One-way Encryption)

- **Definition**: Hash functions take an input and produce a fixed-size string of bytes. They are not reversible, meaning they can't be decrypted back to the original data.
- **Examples**:

a. SHA-256 (Secure Hash Algorithm 256-bit):

- **Algorithm**: SHA-256 generates a 256-bit hash value from the input. It's used in cryptocurrencies (like Bitcoin) and for verifying data integrity.
- **Strength**: Highly secure, resistant to collisions (two inputs generating the same hash).
- **Decryption**: Not possible, as hashing is a one-way operation.

b. MD5 (Message Digest 5):

- **Algorithm**: MD5 produces a 128-bit hash value from any input. Although widely used in the past, it is now considered insecure due to vulnerabilities that allow for hash collisions.
- **Strength**: Fast but insecure for cryptographic purposes.

- **Decryption**: MD5 cannot be decrypted, but it can be brute-forced if the hash is weak.

4. Hybrid Encryption

- **Definition**: Hybrid encryption combines the benefits of both symmetric and asymmetric encryption by using asymmetric encryption to exchange a symmetric key, which is then used for data encryption.
- **Example**:

a. SSL/TLS (Secure Sockets Layer / Transport Layer Security):

- **Algorithm**: SSL/TLS protocols use a combination of asymmetric encryption (usually RSA or ECC) to securely exchange a symmetric key (e.g., AES), which is then used to encrypt the bulk of the data.
- **Strength**: Strong, widely used in securing internet communications (e.g., HTTPS).

5. Stream Ciphers

- **Definition**: Stream ciphers encrypt data one bit or byte at a time.
- **Examples**:

a. RC4 (Rivest Cipher 4):

- **Algorithm**: RC4 generates a stream of pseudo-random bits (keystream) that are XORed with the plaintext to generate ciphertext.
- **Strength**: Fast, but considered insecure due to weaknesses in the algorithm.

- **Decryption**: The ciphertext is XORed with the same keystream to recover the plaintext.

Summary of Algorithm Types:

1. **Symmetric Encryption** (Same key for encryption/decryption): AES, DES, 3DES.
2. **Asymmetric Encryption** (Public and private key): RSA, ECC.
3. **Hash Functions** (One-way, non-reversible): SHA-256, MD5.
4. **Hybrid Encryption** (Combines symmetric and asymmetric encryption): SSL/TLS.
5. **Stream Ciphers** (Bit-by-bit encryption): RC4.

RSA and Caesar Cipher: Overview and Comparison

1. RSA (Rivest-Shamir-Adleman)

Type: Asymmetric Encryption

Key Characteristics:

- **Key Length**: Typically 2048 bits or longer for security.
- **Key Generation**: Involves two keys: a public key (used for encryption) and a private key (used for decryption).
- **Mathematics**: Based on the mathematical properties of prime numbers and the difficulty of factoring large composite numbers.
- **Encryption/Decryption**:
  - To encrypt a message mmm using a public key (e,n)(e, n)(e,n): c≡memod nc \equiv m^e \mod nc≡memodn
  - To decrypt a ciphertext ccc using a private key (d,n)(d, n)(d,n): m≡cdmod nm \equiv c^d \mod nm≡cdmodn

Use Cases:

- Secure data transmission (e.g., SSL/TLS).
- Digital signatures for authentication and integrity.

Strengths:

- Strong security due to large key sizes and the difficulty of factoring.
- Supports secure key exchange and digital signatures.

Weaknesses:

- Slower than symmetric algorithms due to complex calculations.
- Requires more computational resources.

---

2. Caesar Cipher

Type: Symmetric Encryption

Key Characteristics:

- **Key Length**: Typically, a single integer representing the shift value.
- **Key Generation**: The key is a simple integer (the number of positions each letter is shifted).
- **Mathematics**: A basic substitution cipher where each letter in the plaintext is shifted by a fixed number of positions in the alphabet.
- **Encryption/Decryption**:
    - To encrypt a letter ppp with a shift of kkk: $c=(p+k) \mod 26$ $c = (p + k) \mod 26$ $c=(p+k) \mod 26$

- To decrypt a letter ccc: $p=(c-k) \mod 26$ $p = (c - k) \mod 26$ $p=(c−k)mod26$

Use Cases:

- Basic encryption in educational contexts or puzzles.
- Historical significance as one of the earliest known ciphers.

Strengths:

- Simple and easy to implement.
- Fast encryption and decryption.

Weaknesses:

- Very weak security; easily broken by frequency analysis or brute force attacks.
- Only suitable for very simple use cases.

Comparison of RSA and Caesar Cipher

Feature	RSA	Caesar Cipher
Type	Asymmetric	Symmetric
Key Length	2048 bits or more	Typically, a single integer
Key Generation	Public and private keys	Single integer shift value
Encryption Method	Exponentiation and modular arithmetic	Simple letter shifting

Feature	RSA	Caesar Cipher
Strength	Very strong, relies on factoring	Weak, easily broken
Use Cases	Secure communications, digital signatures	Simple encryption, puzzles
Speed	Slower due to complex calculations	Fast and efficient
Security Level	High, requires significant computation	Low, can be easily cracked
Scalability	Highly scalable for large data	Not scalable for large data

Summary

- **RSA** is a modern encryption method that provides strong security for sensitive data and is widely used in secure communications. It is more complex but significantly more secure than the **Caesar cipher**, which is a simple and historical encryption method that is easily broken.
- **Caesar cipher** is primarily of educational and historical interest today due to its simplicity and vulnerability to attacks.

Solution

Caesar cipher decryption

```
def shift(liste, a):
    a = a % len(liste);
    return liste[a:] + liste[:a];
```

This provides us the decryption program for the coordinates. Which will be:

09°64'91.5 "N 3°78'77.7 'W'

RSA decryption

Then used the RSA method to decrypt the message :

```
def inverse(a, m):
    for i in range(1, m):
        if (a * i) % m == 1:
            return i
    return None

def trans(C, d, n):
    return pow(C, d, n)
```

Which returned :

Decrypted C: [76, 101, 32, 116, 114, 233, 115, 111, 114, 32, 101, 115, 116, 32, 99, 97, 99, 104, 233, 32, 115, 111, 117, 115, 32, 108, 97, 32, 112, 108, 97, 113, 117, 101, 32, 114, 111, 117, 103, 101]

Decrypted message: **Le trésor est caché sous la plaque rouge.**

Conclusion

In conclusion, RSA is highly secure and widely applicable in real-world scenarios like online banking, secure communications, and authentication, while the Caesar cipher remains primarily a learning tool or puzzle due to its simplicity.