RSA CYPHER MACHINE-MASTER

Start

TEAM MEMBERS




CHIRAG BAGHEL(22BCS031)

JAGRUTH DANGI(22BCS048)

SHARATH MEDIJALA(22BCS069)

M KARTHIKEYA(22BCS073)

PROJECT PURPOSE

The RSA Cipher Machine project demonstrates the use of RSA encryption and decryption in a cryptographic application. It includes generating RSA keys, encrypting messages, decrypting ciphertexts, and managing prime numbers used for cryptographic operations.

DIRECTORY EXPLANATION

The project directory contains:

1. Lib: Core scripts for RSA operations (encryption, decryption, key generation).
2. Primes: Scripts and data for prime number generation.
3. User: Stores user information for the application.
4. Helpmenu.txt and README_simplified.md: Guides for using the project.

CODE EXPLANATION

- Key scripts:
  1. enigma.py: Main script for RSA encryption/decryption.
  2. mathRSA.py: Performs mathematical operations for RSA.
  3. primeGen.py: Generates prime numbers.

- To run the project:
  1. Navigate to the lib/ directory.
  2. Execute: python3 enigma.py
  3. Use commands like 'encrypt', 'decrypt', 'newKey'.

RESULTS SCREENSHOTS

1. RSA Key Generation: Displays public and private keys.

```
@enigma > myKey
(9701979979633, 13707539)
```

```
@enigma > newKey
(939818102426 3, 11377057)
```

2. Encryption: Converts plaintext to ciphertext.

3. Decryption: Recovers original plaintext from ciphertext.

```
@enigma > encrypt (9398181024263, 11377057) iiitdharwad
encryped message:
z!scHKqà}:kb`Ö<---THE MESSAGE ENDS HERE

(end char ("Ö") doesn't belong to coded msg)
@enigma > decrypt z!scHKqà}:kb`
iiitdharwad
```

CONCLUSION

From this project, we learn the fundamentals of cryptography, the RSA algorithm, key generation, and encryption/decryption. It highlights the role of prime numbers, modular arithmetic, and the importance of securing digital communication in real-world applications.

THANK YOU!