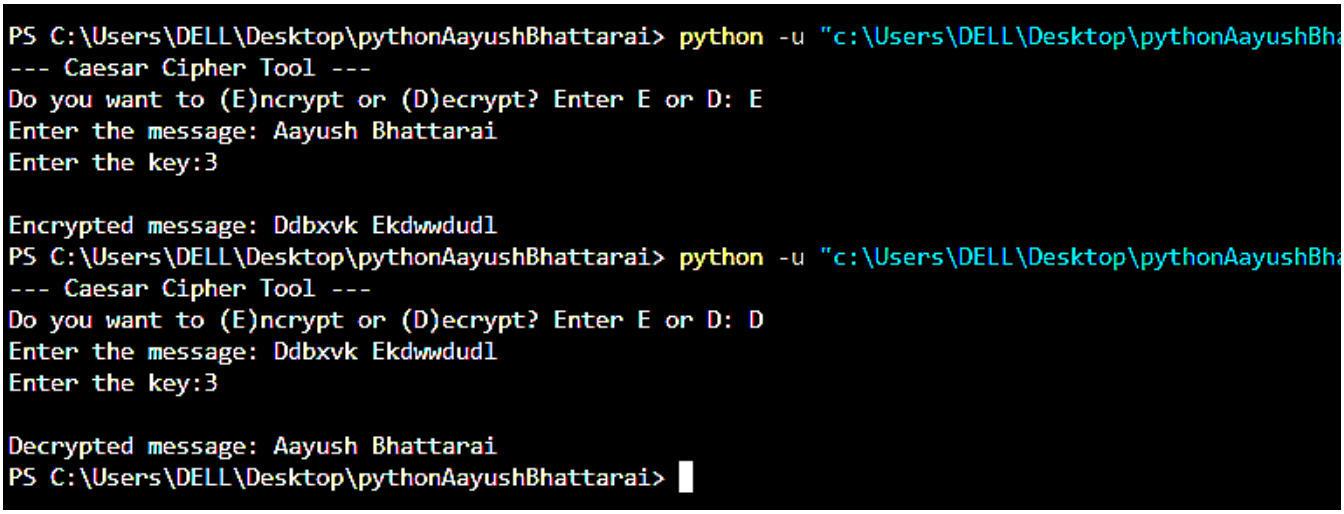
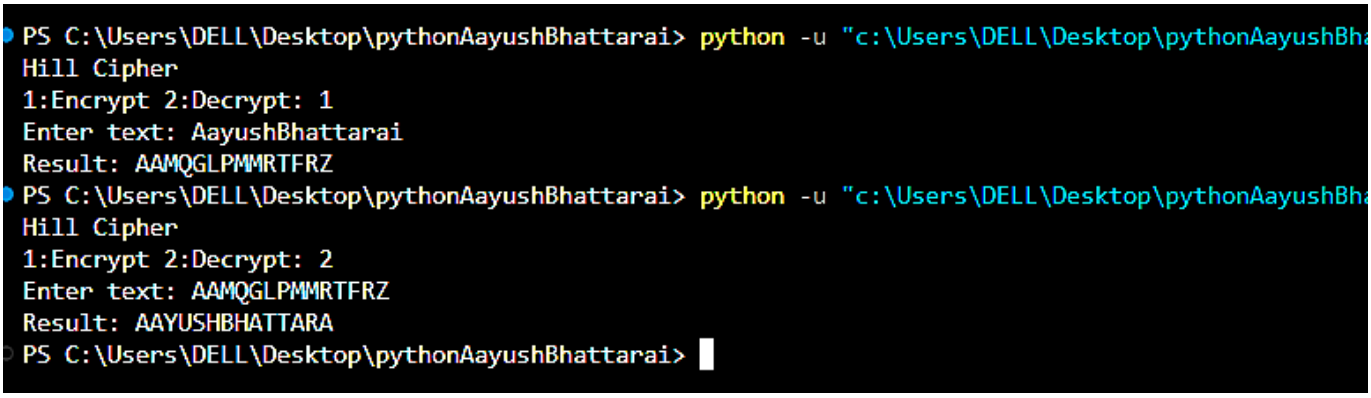
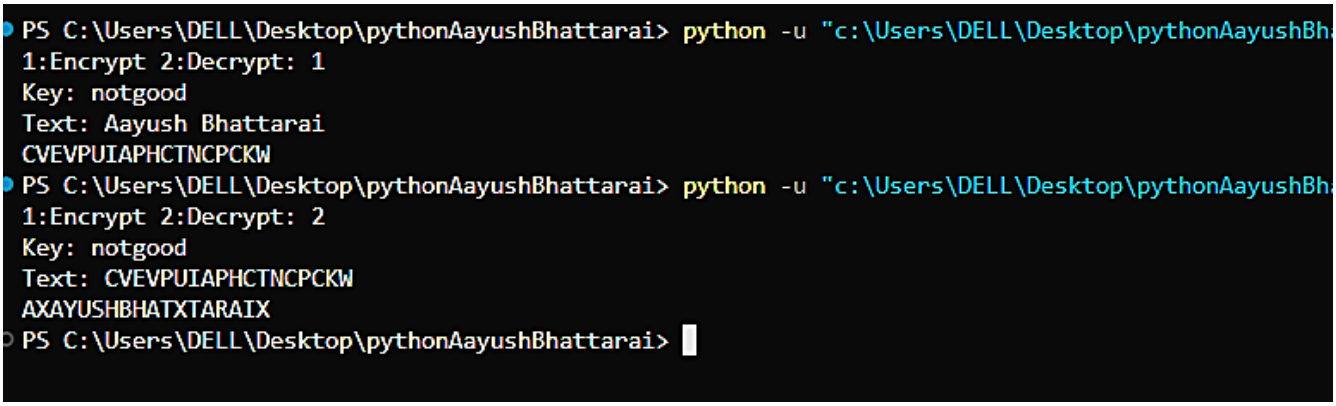
Output:


PS C:\Users\DELL\Desktop\pythonAayushBhattarai> python -u "c:\Users\DELL\Desktop\pythonAayushBha
--- Caesar Cipher Tool ---
Do you want to (E)ncrypt or (D)ecrypt? Enter E or D: E
Enter the message: Aayush Bhattarai
Enter the key:3
Encrypted message: Ddbxvk Ekdwwdudl
PS C:\Users\DELL\Desktop\pythonAayushBhattarai> python -u "c:\Users\DELL\Desktop\pythonAayushBha
--- Caesar Cipher Tool ---
Do you want to (E)ncrypt or (D)ecrypt? Enter E or D: D
Enter the message: Ddbxvk Ekdwwdudl
Enter the key:3
Decrypted message: Aayush Bhattarai
PS C:\Users\DELL\Desktop\pythonAayushBhattarai>

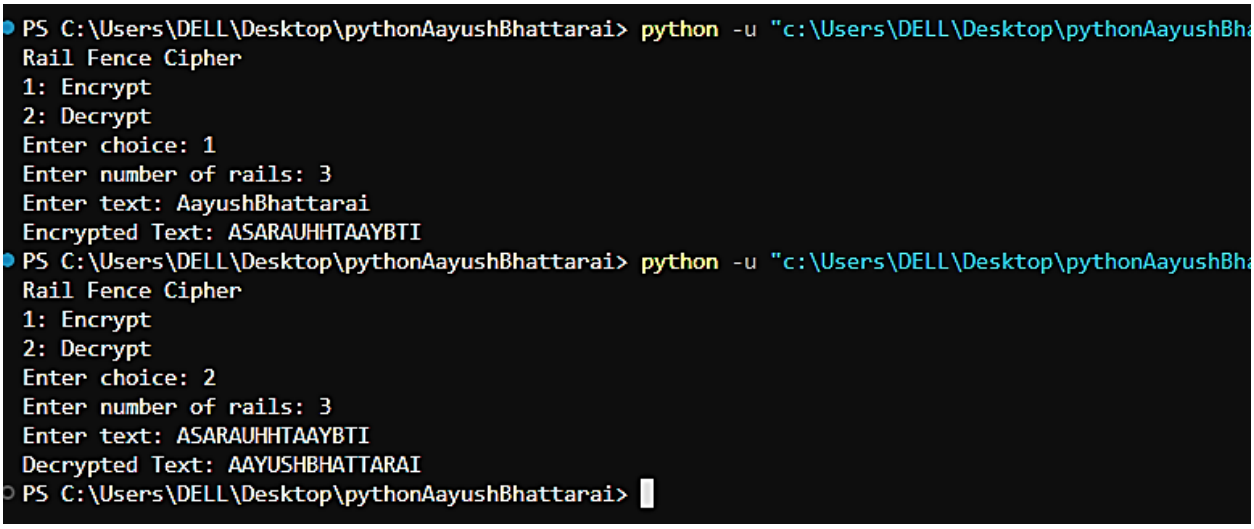
Output:


PS C:\Users\DELL\Desktop\pythonAayushBhattarai> python -u "c:\Users\DELL\Desktop\pythonAayushBha
Hill Cipher
1:Encrypt 2:Decrypt: 1
Enter text: AayushBhattarai
Result: AAMQGLPMMRTFRZ
PS C:\Users\DELL\Desktop\pythonAayushBhattarai> python -u "c:\Users\DELL\Desktop\pythonAayushBha
Hill Cipher
1:Encrypt 2:Decrypt: 2
Enter text: AAMQGLPMMRTFRZ
Result: AAYUSHBHATTARA
PS C:\Users\DELL\Desktop\pythonAayushBhattarai>

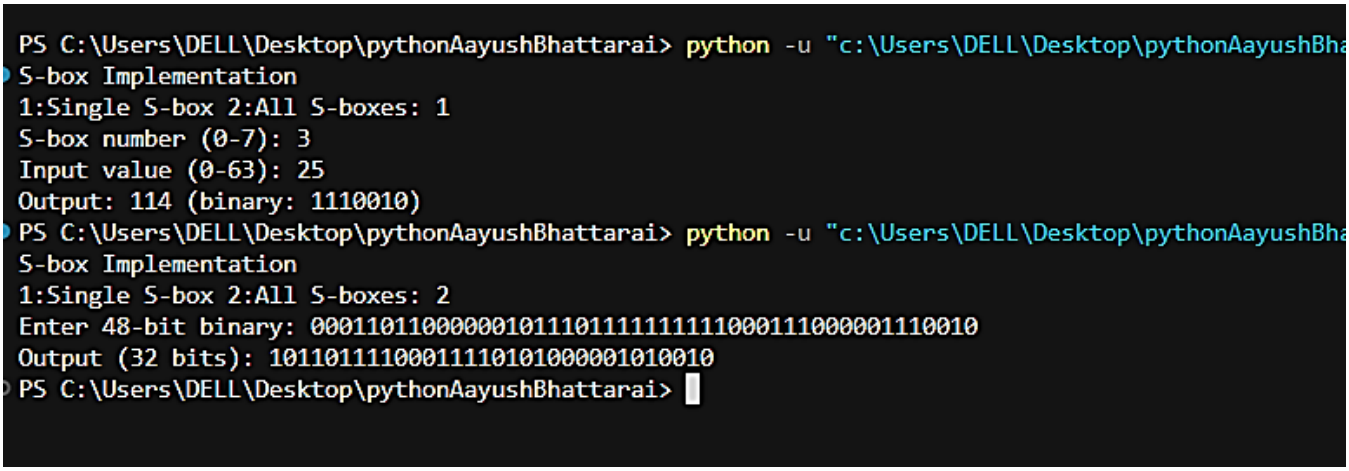
Output:


PS C:\Users\DELL\Desktop\pythonAayushBhattarai> python -u "c:\Users\DELL\Desktop\pythonAayushBh
1:Encrypt 2:Decrypt: 1
Key: notgood
Text: Aayush Bhattarai
CVEVPUIAPHCTNCPCKW
PS C:\Users\DELL\Desktop\pythonAayushBhattarai> python -u "c:\Users\DELL\Desktop\pythonAayushBh
1:Encrypt 2:Decrypt: 2
Key: notgood
Text: CVEVPUIAPHCTNCPCKW
AXAYUSHBHATXTARAIX
PS C:\Users\DELL\Desktop\pythonAayushBhattarai>

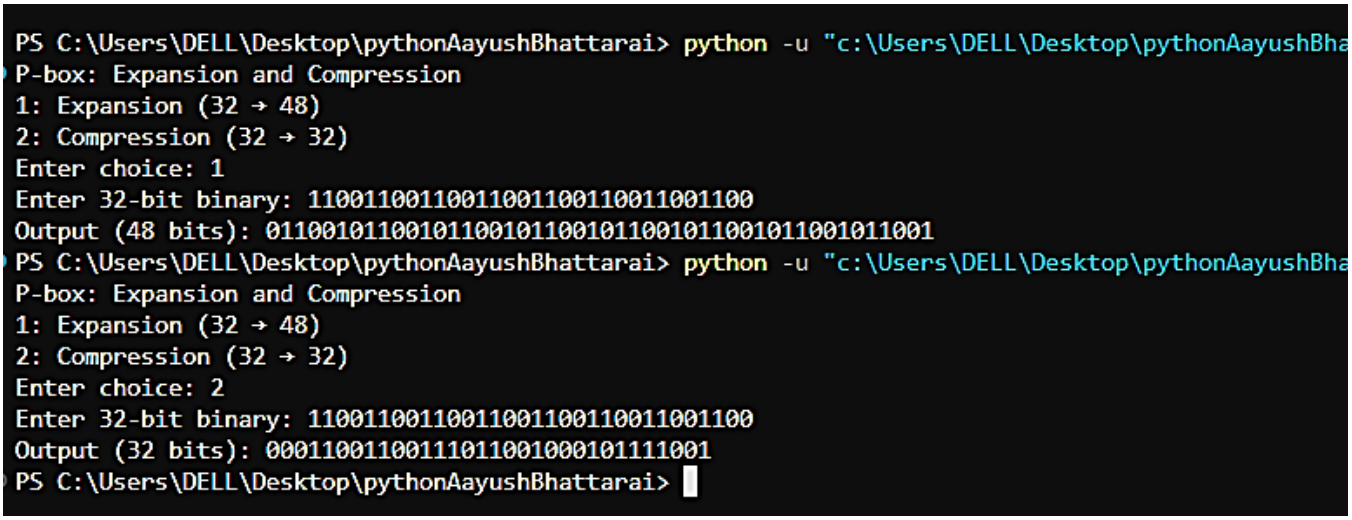
Output:


PS C:\Users\DELL\Desktop\pythonAayushBhattarai> python -u "c:\Users\DELL\Desktop\pythonAayushBha
Rail Fence Cipher
1: Encrypt
2: Decrypt
Enter choice: 1
Enter number of rails: 3
Enter text: AayushBhattarai
Encrypted Text: ASARAUHHTAAYBTI
PS C:\Users\DELL\Desktop\pythonAayushBhattarai> python -u "c:\Users\DELL\Desktop\pythonAayushBha
Rail Fence Cipher
1: Encrypt
2: Decrypt
Enter choice: 2
Enter number of rails: 3
Enter text: ASARAUHHTAAYBTI
Decrypted Text: AAYUSHBHATTARAI
PS C:\Users\DELL\Desktop\pythonAayushBhattarai>

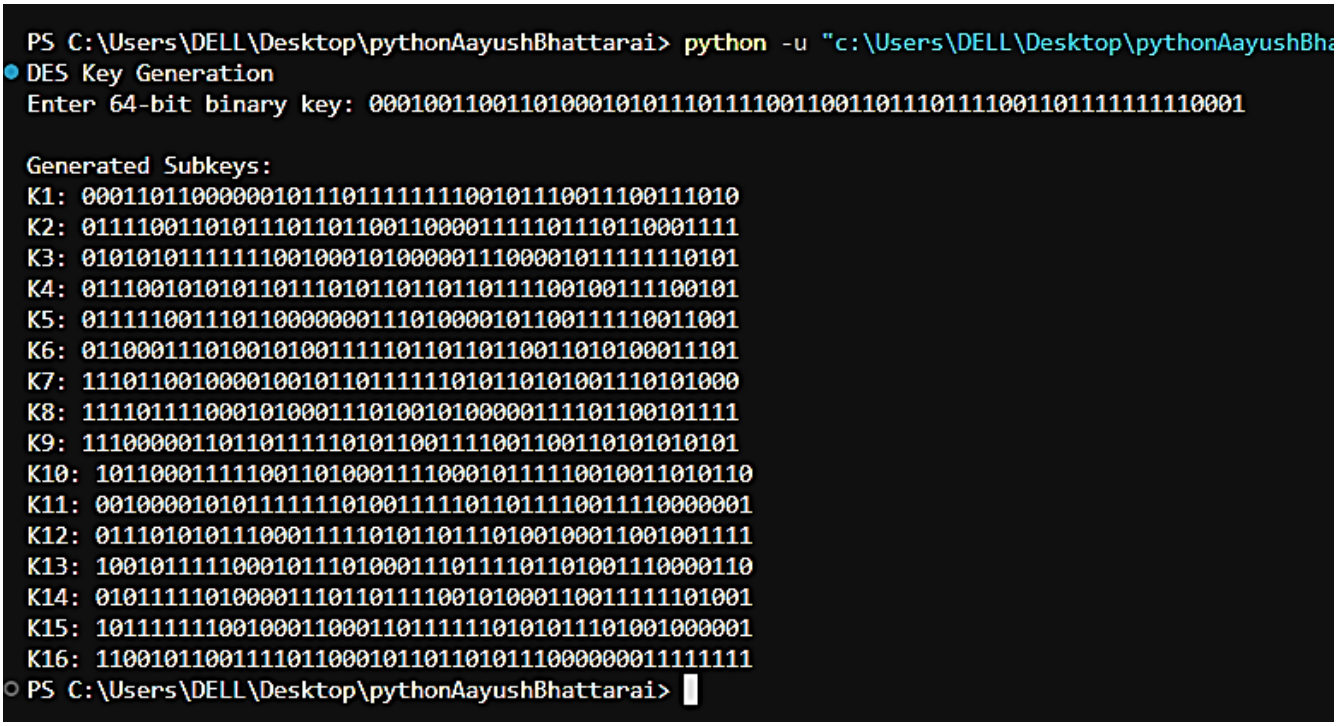
Output:


PS C:\Users\DELL\Desktop\pythonAayushBhattarai> python -u "c:\Users\DELL\Desktop\pythonAayushBha
S-box Implementation
1:Single S-box 2:All S-boxes: 1
S-box number (0-7): 3
Input value (0-63): 25
Output: 114 (binary: 1110010)
PS C:\Users\DELL\Desktop\pythonAayushBhattarai> python -u "c:\Users\DELL\Desktop\pythonAayushBha
S-box Implementation
1:Single S-box 2:All S-boxes: 2
Enter 48-bit binary: 000110110000001011101111111111000111000001110010
Output (32 bits): 10110111100011110101000001010010
PS C:\Users\DELL\Desktop\pythonAayushBhattarai>

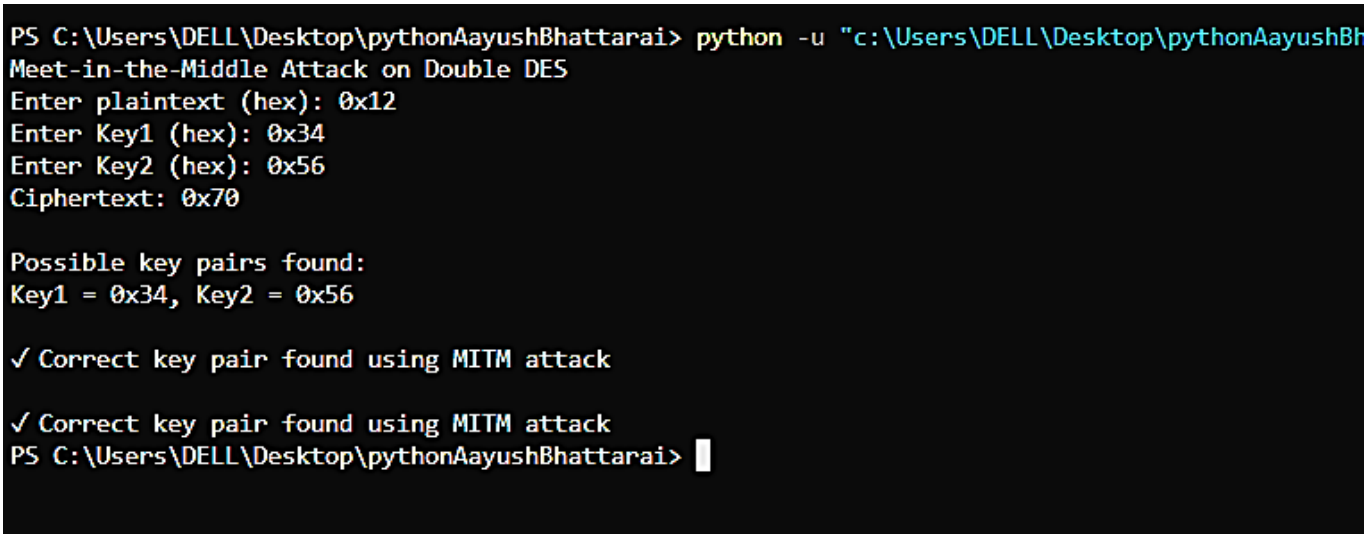
Output:


PS C:\Users\DELL\Desktop\pythonAayushBhattarai> python -u "c:\Users\DELL\Desktop\pythonAayushBha
P-box: Expansion and Compression
1: Expansion (32 → 48)
2: Compression (32 → 32)
Enter choice: 1
Enter 32-bit binary: 11001100110011001100110011001100
Output (48 bits): 011001011001011001011001011001011001011001011001
PS C:\Users\DELL\Desktop\pythonAayushBhattarai> python -u "c:\Users\DELL\Desktop\pythonAayushBha
P-box: Expansion and Compression
1: Expansion (32 → 48)
2: Compression (32 → 32)
Enter choice: 2
Enter 32-bit binary: 11001100110011001100110011001100
Output (32 bits): 00011001100111011001000101111001
PS C:\Users\DELL\Desktop\pythonAayushBhattarai>

Output:


PS C:\Users\DELL\Desktop\pythonAayushBhattarai> python -u "c:\Users\DELL\Desktop\pythonAayushBha
DES Key Generation
Enter 64-bit binary key: 0001001100110100010101110111100110011011101111001101111111110001
Generated Subkeys:
K1: 000110110000001011101111111100101110011100111010
K2: 011110011010111011011001100001111101110110001111
K3: 010101011111111001000101000001110000101111111010 1
K4: 011100101010111011010110110111110010011110010 1
K5: 011111001110110000000111010000101100111110011001
K6: 011000111010010100111110111011011001101010001 1101
K7: 111011001000010010111011111101011010100111010 1000
K8: 111101111000101000111010010100000111101100101111
K9: 111000001101101111101011001111001100110101010101
K10: 101100011111001101000111100010111110010011010110
K11: 001000010101111111010011111011011110011110000001
K12: 011101010111000111110101101110100100011001001111
K13: 100101111100010111010001110111101101001110000110
K14: 010111110100001110110111100101000110011111101001
K15: 101111111001000110001101111110101011101001000001
K16: 110010110011110110001011011010111000000011111111
PS C:\Users\DELL\Desktop\pythonAayushBhattarai>

Output:


PS C:\Users\DELL\Desktop\pythonAayushBhattarai> python -u "c:\Users\DELL\Desktop\pythonAayushBh
Meet-in-the-Middle Attack on Double DES
Enter plaintext (hex): 0x12
Enter Key1 (hex): 0x34
Enter Key2 (hex): 0x56
Ciphertext: 0x70
Possible key pairs found:
Key1 = 0x34, Key2 = 0x56
✓ Correct key pair found using MITM attack
✓ Correct key pair found using MITM attack
PS C:\Users\DELL\Desktop\pythonAayushBhattarai>