###### Experiment Number: 01

**TITLE: Encryption Method Algorithms**

**PROBLEM STATEMENT:** Write a program in python/ Java/ Scala/ C++/ HTML5 to implement password data encryption. Use encryption method overloading (any two methods studied)

**OBJECTIVES:**

1. To develop a good understanding of the working of various block and stream cipher encryption algorithms
2. To understand detailed working and implementation of Playfair cipher and Caeser cipher algorithms in python. Compare it to the working of DES and AES

**THEORY:**

**Algorithm of Caeser cipher:**

1. Read each character in the plaintext
2. Substitute each character in the plaintext with a character three places down the line
3. Repeat the process for each character in the plaintext

**Algorithm of Playfair cipher:**

1. Creation and population of key matrix
   1. Create a 5x5 matrix, enter the keyword in the matrix row-wise left-to-right and then from top-to-bottom
   2. Drop duplicate letters
   3. Fill in the rest of the matrix with the rest of the English alphabets which were not part of the keyword. While doing so combine I and J in part of the same cell
2. Encrypt
   1. Break the plaintext message into pairs
   2. If both the alphabets of the plaintext message are the same or only one alphabet is left, replace the second one with X
   3. If both the alphabets in the pair appear in the same row of the key matrix, replace them with the alphabets to their immediate right in the matrix
   4. If both the alphabets in the pair appear in the same column of the key matrix, replace them with the alphabets immediately below them in the matrix
   5. If the alphabets are not in the same row or column, replace them with the alphabets in the same row respectively but at the other pair of corner of the rectangle defined by the original pair

**MATHEMATICAL MODEL:**

**Caeser Cipher:**

1. **Input:**

I={Plaintext byte,key}

1. **Output**:

**O={**Ciphertext byte}

1. **Process**:

* Plaintext: ABCDEFGHIJKLMNOPQRSTUVWXYZ
* Ciphertext: DEFGHIJKLMNOPQRSTUVWXYZABC

1. **Mathematically**

Encryption of a letter xby a shift *n* can be described mathematically as

https://upload.wikimedia.org/math/b/b/b/bbb819c72cda43180d98e6ade5cadb04.png

Decryption id performed as

D_n(x) = (x - n) \mod {26}.

1. Let U={I,O,F,S,T}

I= {I1,I2}

I1= {Plaintext Alphabet, Plaintext Number}

I2={Number of Shifts}

O={Ciphertext alphabet, ciphertext number}

S= Case of success where the plaintext is successfully encrypted

F= Case of failure where the plaintext given is invalid

T= { encryption(plain\_text,key), decryption(cipher\_text,key) }

State diagram:

x: plaintext character

n: number of shifts

x, n https://upload.wikimedia.org/math/b/b/b/bbb819c72cda43180d98e6ade5cadb04.png ciphertext

**Playfair Cipher:**

1. **Input:**

**I={**Plaintext byte, keyword}

1. **Output:**

O={Ciphertext byte}

1. **Process:**
   1. 5x5 matrix using the keyword is constructed
   2. Divide plaintext in pairs
   3. Same row but different columns

Plaintext=>(aij,aik) Ciphertext=>(ai(j+1),ai(k+1))

* 1. Different row but same column

Plaintext=>(aij,ajk) Ciphertext=>(a(i+1)j,a(j+1)k)

* 1. Different rows and different columns

Plaintext=>(aij,amn) Ciphertext=>(ain,amj)

1. Let U={I,O,F,S,T}

I= {I1,I2}

I1= {Plaintext}

I2={Key}

O={Ciphertext}

S= Case of success where the plaintext is successfully encrypted

F= Case of failure where the plaintext given is invalid

T= { encryption(plain\_text,key), decryption(cipher\_text,key) }

State diagram:

Same row

P, K 5x5 matrix, P in pairs

Same column

Different row,

different column

Ciphertext

**IMPLEMENTATION DETAILS / DESIGN LOGIC:**

*(Algorithm/Flow Charts/Pseudo Code/DFD/UML diagrams)*

**Algorithm:-**

Caeser Cipher: Playfair cipher

Accept plaintext byte and key n

Accept plaintext and key

Replace plaintext with nth letter down

Construct a 5x5 matrix with the key

Break the plaintext in pairs

Check position of plaintext pairs

Different row and column, replace with the alphabets in the same row respectively but at the other pair of corner of the rectangle defined by the original pair

Same row, replace with immediate right letter

Same column, replace with letter immediately below

**Input :-** Plaintext byte, key”

Plain\_text = “HELLO”

Key = “CIPHER”

**Expected Output :-** Ciphertext byte

Cipher text = ECSPGS

**Execute the program with the following commands.**

python3 caeser\_cipher.py

python3 playfair.py

**TEST CASES:**

|  |  |
| --- | --- |
| **Test case Input** | **Test case Output** |
| Accept a plaintext(alphabets) along with key | Cipher text generated successfully |
| Accept a plaintext(special characters) along with key | Cipher text should not be generate, error generated |

**CONCLUSION:**

We have successfully implemented caeser cipher and playfair cipher substitution methods.

**COURSE OUTCOMES ACHIEVED:**

|  |  |
| --- | --- |
| **COURSE OUTCOME** | **ACHEIVED** |
| To solve problems using mathematical modeling | **√** |
| To use software design methods and testing |  |
| To solve problems for multicore or distributed, concurrent/parallel environments | **√** |

**FAQ’s**

1. What is the difference between block cipher and stream cipher?
   1. Stream cipher technique involves the encryption of one plaintext byte at a time.
   2. Block cipher technique involves the encryption of one block at a time which is a sequence of plaintext bytes at a time
2. What is the disadvantage in caeser cipher?
   1. Caeser cipher is a substitution technique which directly replaces the plaintext letter with ‘n’ letters down the alphabet range. It is very easy to decrypt if the difference between a plaintext letter and the ciphertext letter is known. Hence it is not used practically