**Different Cipher methods used for encryption and decryption.**

The two main categories of cypher are as follows:

**Substitution Ciphers** – This works by replacing letters or a group of letters with another letter or symbol.

Methods used in Substitution Ciphers –

1. Caesar Cipher:

|  |
| --- |
| Shift = 3 |
| Example:  **1.Encrypt**  PT = ‘HELLO’  CT = ‘KHOOR’  **2.Decrypt**  PT = ‘KHOOR’  CT = ‘HELLO’ |

1. Monoalphabetic Substitution cipher:

|  |
| --- |
| PT Alphabets: ‘ABCDEFGHIJKLMNOPQRSTUVWXYZ’ |
| CT Alphabets: ‘QWERTYUIOPASDFGHJKLZXCVBNM’ |
| Example:  **1.Encrypt**  PT = ‘HELLO’  CT = ‘TUSSR’  **2.Decrypt**  PT = ‘TUSS’  CT = ‘HELLO’ |
|  |

1. One-time pad cipher:

|  |
| --- |
| Key should be random each time and will be symmetric,  Also it should be of the same length as of the original string. |
| A = 0, B = 1, C = 2, ………………Z = 25 |
| Example:  PT = ‘HELLO’  Key = ‘WXYZL’  H(7) + W(22) = 29 -> (29%26) = 3 -> D  E(4) + X(23) = 27 -> (27%26) = 1 ->B  L(11) + Y(24) = 35 -> (35%26) = 9 ->J  L(11) + Z(25) = 36 ->(36%26) = 10 -> K  O(14) + L(11) = 25 -> (25%26) = 25 -> Z  CT = ‘DBJKZ’ |

1. Atbash Cipher:

|  |
| --- |
| PT Alphabets: ‘ABCDEFGHIJKLMNOPQRSTUVWXYZ’  CT Alphabets: ‘ZYXWVUTSRQPONMLKJIHGFEDCBA’ |
| Example:  **1.Encrypt**  PT = ‘HELLO’  CT = ‘SVOOL’  **2.Decrypt**  PT = ‘SVOOL’  CT = ‘HELLO’ |
|  |

1. Affine Cipher:

|  |
| --- |
| Encryption function: E(x) = (ax + b) mod m  where,  x – numerical value of PT letter  a and b – key values of the cipher  m – size of the alphabet i.e 26 |
| A = 0, B = 1, C = 2, ………………Z = 25 |
| Example:  PT = ‘HELLO’  H = 7, E = 4, L = 11, L =11, O = 14  let a = 2, b =4, m =26  E(H) = (2\*7 + 4) % 26 = 18 -> S  E(E) = (2\*4 + 4) % 26 = 12 -> M  E(L) = (2\*11 + 4) % 26 = 0 -> A  E(L) = (2\*11 + 4) % 26 = 0 -> A  E(O) = (2\*14 + 4) % 26 = 6 -> G  CT = ‘SMAAG’ |

1. ROT13 Cipher:

|  |
| --- |
| Shift = 13 |
| Example:  **1.Encrypt:**  PT = ‘HELLO’  CT = ‘URYYB’  **2.Decrypt:**  PT = ‘URYYB’  CT = ‘HELLO’ |

1. Vigenere Cipher:

|  |
| --- |
| Conversion can be done using Viganere table or the formula which is given by  Ci = (Pi + Ki) mod 26  where,  Pi – Position of the PT letter  Ki – Position of Keyword letter  Ci – position of CT letter |
| Example:  PT = ‘HELLO’  Keyword = ‘KEYKE’  H (7) + K (10) = 17 → R  E (4) + E (4) = 8 → I  L (11) + Y (24) = 35 → 9 → J  L (11) + K (10) = 21 → V  O (14) + E (4) = 18 → S  CT = ‘RIJVS’ |

1. Keyword Cipher:

|  |
| --- |
| Example:  Keyword = Keyword  PT Alphabet = ‘ABCDEFGHIJKLMNOPQRSTUVWXYZ’  CT Alphabet = ‘KEYWORDABCFGHIJLMNPQSTUVXZ’  PT = ‘HELLO’  CT = ‘AOGGL’ |

1. Homophonic Substitution Cipher:

|  |
| --- |
| Example:  PT = ‘HELLO’  Mapping: It is done randomly  H -> {1,2}  E -> {3,4}  L -> {5,6}  O -> {7,8}  CT = 13567 or 24568 or so on |

1. Alberti Cipher:

|  |
| --- |
| Disk1 Alphabets (PT Alphabets): ‘ABCDEFGHIJKLMNOPQRSTUVWXYZ’  Disk2 Alphabets (CT Alphabets) in starting alignment: ‘MNOPQRSTUVWXYZABCDEFGHIJKL’ |
| Example:  PT = ‘HELLO’  CT = ‘UQXXA’ |

**Different tools available for substitution methods.**

* **Online Tools**

Cryptii, Boxentriq Substitution Cipher Tool, dCode, CrypTool

* **Programming libraries & Softwares**

PyCryptodome, Cryptography, Mathematica, MATLAB, Microsoft Excel

* **Manual Tools**

Cipher Wheels, Cipher Tables

1. Transposition Ciphers – This works by maintaining the original letters but the original letters will be shuffled and the original message or code will get jumbled so that it cannot be read by unauthorised members or system.

Methods used in transposition Ciphers -

* Columnar Transposition
* Scytale
* Bifid cipher
* Trifid cipher
* ADFGVX cipher
* VIC cipher
* 3 Rail Fence
* 4 Rail Fence
* Columnar Transposition
* Permutation
* Double Transposition Cipher
* Simple Columnar Transposition
* Double Columnar Transposition
* Four-Square
* Cipher Block Transposition
* Columnar Transposition with Reversed Keyword
* Diagonal Transposition
* Triangular Transposition
* Inverted Columnar
* Reverse Columnar
* Row Transposition
* Transposition Cipher with Random Permutation