**Affine cipher**

The **affine cipher** is a type of **monoalphabetic substitution cipher**, wherein each letter in an alphabet is mapped to its numeric equivalent, encrypted using a simple mathematical function, and converted back to a letter. The formula used means that each letter encrypts to one other letter, and back again, meaning the cipher is essentially a standard substitution cipher with a rule governing which letter goes to which. As such, it has the weaknesses of all substitution ciphers.

Each letter is enciphered with the function (ax+b)\mod(26), where bis the magnitude of the shift.

|  |
| --- |
|  |

**Description**0 .. m-1

In the affine cipher the letters of an alphabet of size **n** are first mapped to the integers in the range 0..n-1. It then uses **modular arithmetic** to transform the integer that each plaintext letter corresponds to into another integer that correspond to a ciphertext letter. The encryption function for a single letter is

\mbox{E}(x)=(ax+b)\mod{m},

where modulus mis the size of the alphabet and aand bare the key of the cipher. The value amust be chosen such that aand mare [coprime](http://en.wikipedia.org/wiki/Coprime). The decryption function is

\mbox{D}(x)=a^{-1}(x-b)\mod{m},

where a^{-1}is the **modular multiplicative inverse** of a[modulo](http://en.wikipedia.org/wiki/Modular_arithmetic) m. I.e., it satisfies the equation

1 = a a^{-1}\mod{m}.

The **multiplicative inverse** of aonly exists if aand mare coprime. Hence without the restriction on adecryption might not be possible. It can be shown as follows that decryption function is the inverse of the encryption function,

\begin{align}
\mbox{D}(\mbox{E}(x)) &= a^{-1}(\mbox{E}(x)-b)\mod{m}\\
  &= a^{-1}(((ax+b)\mod{m})-b)\mod{m} \\
  &= a^{-1}(ax+b-b)\mod{m} \\
  &= a^{-1}ax \mod{m}\\
  &= x\mod{m}.
\end{align}

**Weaknesses**

Since the affine cipher is still a monoalphabetic substitution cipher, it inherits the weaknesses of that class of ciphers. The **Caesar cipher** is the Affine cipher when a=1since the encrypting function simply reduces to a linear shift.

Considering the specific case of encrypting messages in English (i.e. m = 26), there are a total of 286 non-trivial affine ciphers, not counting the 26 trivial Caesar ciphers. This number comes from the fact there are 12 numbers that are coprime with 26 that are less than 26 (these are the possible values of **a**). Each value of **a** can have 26 different addition shifts (the **b** value); therefore, there are 12\*26 or 312 possible keys. This lack of variety renders the system as highly insecure when considered in light of **Kerckhoffs' Principle**.

The cipher's primary weakness comes from the fact that if the cryptanalyst can discover (by means of [frequency analysis](http://en.wikipedia.org/wiki/Frequency_analysis), brute force, guessing or otherwise) the plaintext of two ciphertext characters then the key can be obtained by solving a [simultaneous equation](http://en.wikipedia.org/wiki/Simultaneous_equation). Since we know aand mare relatively prime this can be used to rapidly discard many "false" keys in an automated system.

The same type of transformation used in affine ciphers is used in [linear congruential generators](http://en.wikipedia.org/wiki/Linear_congruential_generator), a type of [pseudorandom number generator](http://en.wikipedia.org/wiki/Pseudorandom_number_generator). This generator is not a [cryptographically secure pseudorandom number generator](http://en.wikipedia.org/wiki/Cryptographically_secure_pseudorandom_number_generator) for the same reason that the affine cipher is not secure.

**Examples**

In these two examples, one encrypting and one decrypting, the alphabet is going to be the letters A through Z, and will have the corresponding values found in the following table.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **A** | **B** | **C** | **D** | **E** | **F** | **G** | **H** | **I** | **J** | **K** | **L** | **M** | **N** | **O** | **P** | **Q** | **R** | **S** | **T** | **U** | **V** | **W** | **X** | **Y** | **Z** |
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 |

**Encrypting**

In this encrypting example,the plaintext to be encrypted is "AFFINE CIPHER" using the table mentioned above for the numeric values of each letter, taking **a** to be 5, **b** to be 8, and **m** to be 26 since there are 26 characters in the alphabet being used. Only the value of **a** has a restriction since it has to be coprime with 26. The possible values that **a** could be are 1, 3, 5, 7, 9, 11, 15, 17, 19, 21, 23, and 25. The value for **b** can be arbitrary as long as **a**  does not equal 1 since this is the shift of the cipher. Thus, the encryption function for this example will be  y=E(x)=(5x+8)\pmod{26}. The first step in encrypting the message is to write the numeric values of each letter.

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| plaintext: | A | F | F | I | N | E | C | I | P | H | E | R |
| x: | 0 | 5 | 5 | 8 | 13 | 4 | 2 | 8 | 15 | 7 | 4 | 17 |

Now, take each value of x, and solve the first part of the equation, (5x+8). After finding the value of (5x + 8) for each character, take the remainder when dividing the result of (5x + 8) by 26. The following table shows the first four steps of the encrypting process.

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| plaintext: | A | F | F | I | N | E | C | I | P | H | E | R |
| x: | 0 | 5 | 5 | 8 | 13 | 4 | 2 | 8 | 15 | 7 | 4 | 17 |
| 5x+8 | 8 | 33 | 33 | 48 | 73 | 28 | 18 | 48 | 83 | 43 | 28 | 93 |
| (5x+8)\pmod{26} | 8 | 7 | 7 | 22 | 21 | 2 | 18 | 22 | 5 | 17 | 2 | 15 |

The final step in encrypting the message is to look up each numeric value in the table for the corresponding letters. In this example, the encrypted text would be IHHWVCSWFRCP. The table below shows the completed table for encrypting a message in the Affine cipher.

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| plaintext: | A | F | F | I | N | E | C | I | P | H | E | R |
| x: | 0 | 5 | 5 | 8 | 13 | 4 | 2 | 8 | 15 | 7 | 4 | 17 |
| 5x+8 | 8 | 33 | 33 | 48 | 73 | 28 | 18 | 48 | 83 | 43 | 28 | 93 |
| (5x+8)\pmod{26} | 8 | 7 | 7 | 22 | 21 | 2 | 18 | 22 | 5 | 17 | 2 | 15 |
| ciphertext: | I | H | H | W | V | C | S | W | F | R | C | P |

**Decrypting**

In this decryption example, the ciphertext that will be decrypted is the ciphertext from the encryption example. The corresponding decryption function is \mbox{D}(y)=21(y-8)\mbox{ mod }26, where a^{-1}is calculated to be 21, bis 8, and mis 26. To begin, write the numeric equivalents to each letter in the ciphertext, as shown in the table below.

a a^{-1}mod 26 = 1

5 x 21 mod 26 = 1

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| ciphertext: | I | H | H | W | V | C | S | W | F | R | C | P |
| y: | 8 | 7 | 7 | 22 | 21 | 2 | 18 | 22 | 5 | 17 | 2 | 15 |

Now, the next step is to compute 21(y-8), and then take the remainder when that result is divided by 26. The following table shows the results of both computations.

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| ciphertext: | I | H | H | W | V | C | S | W | F | R | C | P |
| y: | 8 | 7 | 7 | 22 | 21 | 2 | 18 | 22 | 5 | 17 | 2 | 15 |
| 21(y-8): | 0 | -21 | -21 | 294 | 273 | -126 | 210 | 294 | -63 | 189 | -126 | 147 |
| (21(y-8)) mod 26: | 0 | 5 | 5 | 8 | 13 | 4 | 2 | 8 | 15 | 7 | 4 | 17 |

The final step in decrypting the ciphertext is to use the table to convert numeric values back into letters. The plaintext in this decryption is AFFINECIPHER. Below is the table with the final step completed.

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| ciphertext: | I | H | H | W | V | C | S | W | F | R | C | P |
| y: | 8 | 7 | 7 | 22 | 21 | 2 | 18 | 22 | 5 | 17 | 2 | 15 |
| 21(y-8): | 0 | -21 | -21 | 294 | 273 | -126 | 210 | 294 | -63 | 189 | -126 | 147 |
| (21(y-8)) mod 26: | 0 | 5 | 5 | 8 | 13 | 4 | 2 | 8 | 15 | 7 | 4 | 17 |
| plaintext: | A | F | F | I | N | E | C | I | P | H | E | R |

**Entire alphabet encoded**

To make encrypting and decrypting quicker, the entire alphabet can be encrypted to create a one to one map between the letters of the cleartext and the ciphertext. In this example, the one to one map would be the following:

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **letter in the cleartext** | **A** | **B** | **C** | **D** | **E** | **F** | **G** | **H** | **I** | **J** | **K** | **L** | **M** | **N** | **O** | **P** | **Q** | **R** | **S** | **T** | **U** | **V** | **W** | **X** | **Y** | **Z** |
| number in the cleartext | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 |
| (5x+8)mod(26) | 8 | 13 | 18 | 23 | 2 | 7 | 12 | 17 | 22 | 1 | 6 | 11 | 16 | 21 | 0 | 5 | 10 | 15 | 20 | 25 | 4 | 9 | 14 | 19 | 24 | 3 |
| ciphertext letter | I | N | S | X | C | H | M | R | W | B | G | L | Q | V | A | F | K | P | U | Z | E | J | O | T | Y | D |

**Programming examples**

JAVA:

public void Affine(int a, int b){

for (int num = 0; num < 26; num++)

System.out.println(((char)('A'+num)) + ":" + ((char)('A'+(a\*num + b)% 26)) );

}

Affine(5,8)

Link : <http://en.wikipedia.org/wiki/Affine_cipher#Decrypting>