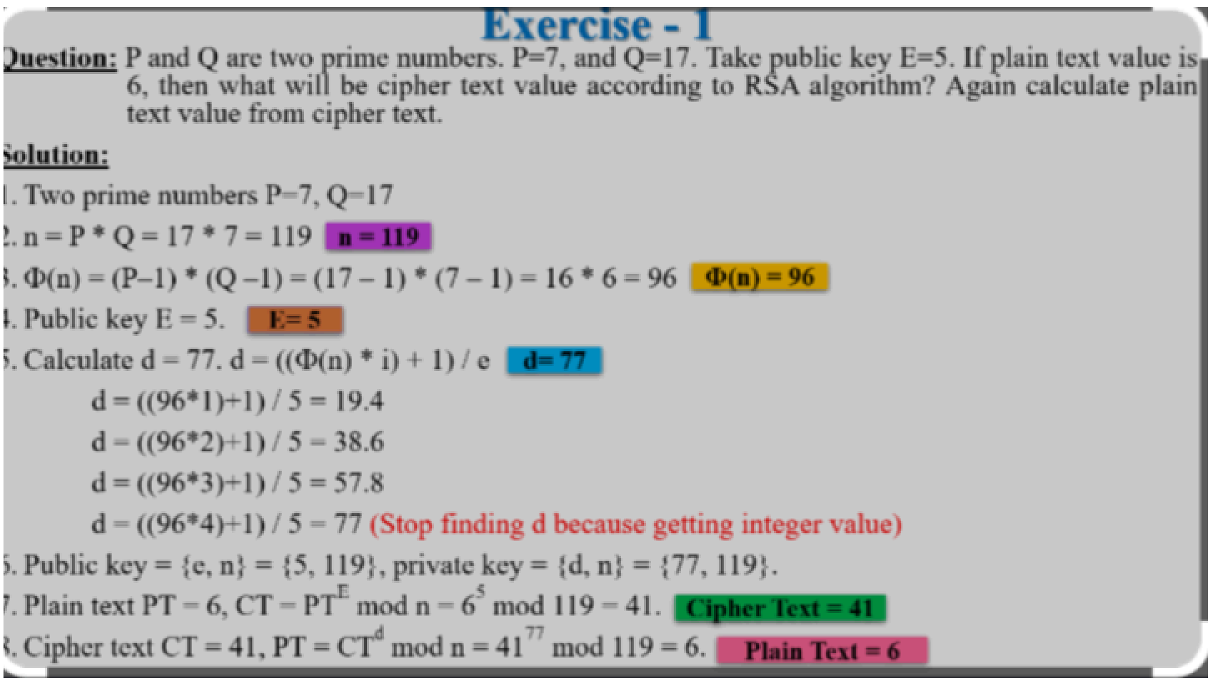
RSA Algorithm Examples:



Exercise - 1

Question: P and Q are two prime numbers. P=7, and Q=17. Take public key E=5. If plain text value is 6, then what will be cipher text value according to RSA algorithm? Again calculate plain text value from cipher text.

Solution:

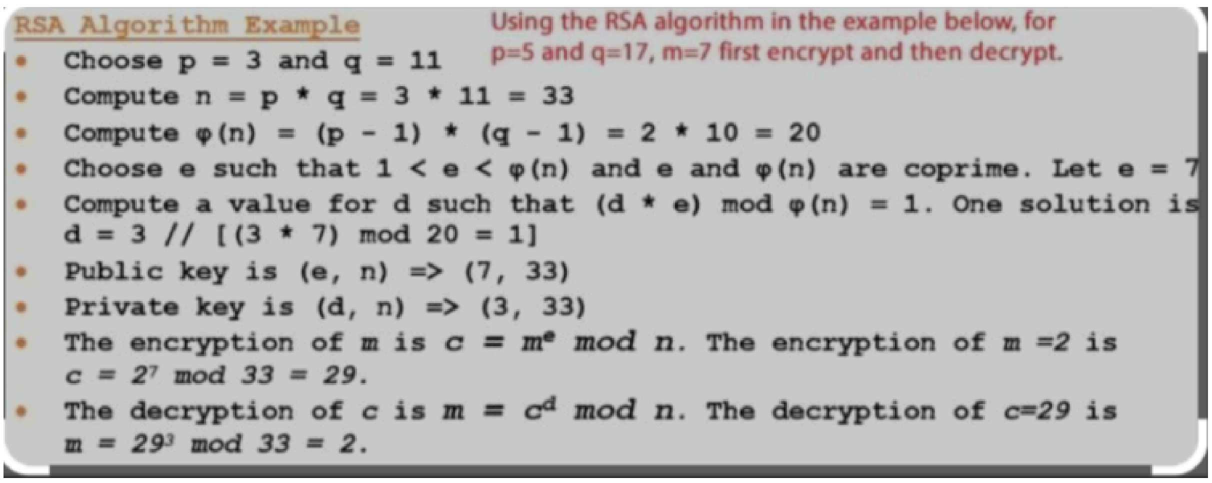
1. Two prime numbers P=7, Q=17
2. $n = P * Q = 17 * 7 = 119$ **n = 119**
3. $\Phi(n) = (P-1) * (Q-1) = (17-1) * (7-1) = 16 * 6 = 96$ **Φ(n) = 96**
4. Public key E = 5. **E= 5**
5. Calculate d = 77. $d = ((\Phi(n) * i) + 1) / e$ **d= 77**

   $d = ((96*1)+1) / 5 = 19.4$

   $d = ((96*2)+1) / 5 = 38.6$

   $d = ((96*3)+1) / 5 = 57.8$

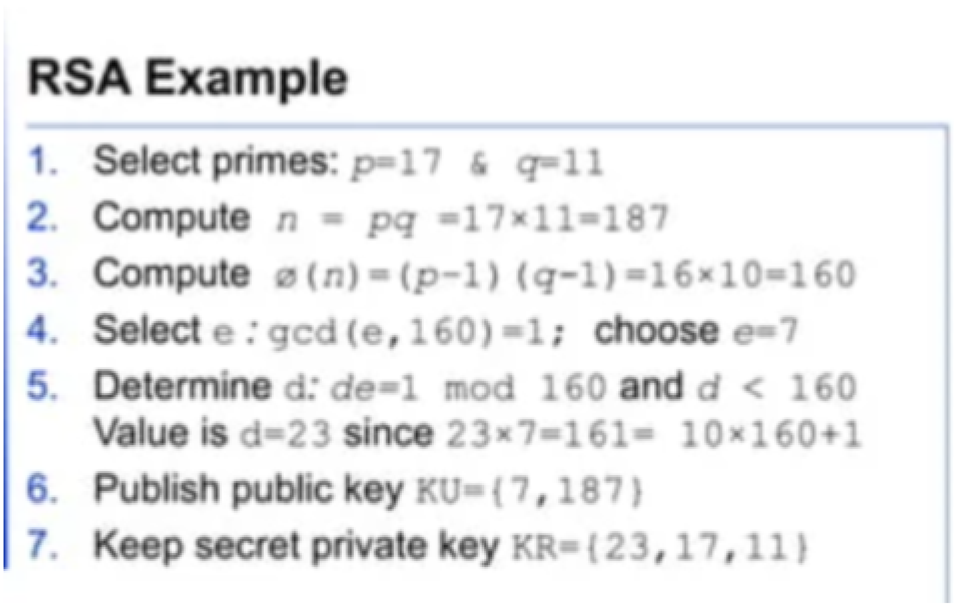
   $d = ((96*4)+1) / 5 = 77$ (Stop finding d because getting integer value)
6. Public key = {e, n} = {5, 119}, private key = {d, n} = {77, 119}.
7. Plain text PT = 6, $CT = PT^E \bmod n = 6^5 \bmod 119 = 41$. **Cipher Text = 41**
8. Cipher text CT = 41, $PT = CT^d \bmod n = 41^{77} \bmod 119 = 6$. **Plain Text = 6**



RSA Algorithm Example

Using the RSA algorithm in the example below, for p=5 and q=17, m=7 first encrypt and then decrypt.

- Choose p = 3 and q = 11
- Compute n = p * q = 3 * 11 = 33
- Compute φ(n) = (p - 1) * (q - 1) = 2 * 10 = 20
- Choose e such that 1 < e < φ(n) and e and φ(n) are coprime. Let e = 7
- Compute a value for d such that (d * e) mod φ(n) = 1. One solution is d = 3 // [(3 * 7) mod 20 = 1]
- Public key is (e, n) => (7, 33)
- Private key is (d, n) => (3, 33)
- The encryption of *m* is $c = m^e \bmod n$. The encryption of *m =2* is $c = 2^7 \bmod 33 = 29$.
- The decryption of *c* is $m = c^d \bmod n$. The decryption of *c=29* is $m = 29^3 \bmod 33 = 2$.



RSA Example

1. Select primes: $p=17$ & $q=11$
2. Compute $n = pq = 17 \times 11 = 187$
3. Compute $\varnothing(n) = (p-1)(q-1) = 16 \times 10 = 160$
4. Select e : $\gcd(e,160)=1$; choose $e=7$
5. Determine d: $de=1 \mod 160$ and $d < 160$
   Value is $d=23$ since $23 \times 7 = 161 = 10 \times 160 + 1$
6. Publish public key $KU=\{7,187\}$
7. Keep secret private key $KR=\{23,17,11\}$

RC4 Encryption Algorithm:-

RC4 is a stream cipher and variable-length key algorithm. This algorithm encrypts one byte at a time (or larger units at a time). A key input is a pseudorandom bit generator that produces a stream 8-bit number that is unpredictable without knowledge of input key, The output of the generator is called key-stream, is combined one byte at a time with the plaintext stream cipher using X-OR operation.

Example

```
RC4 Encryption
10011000 ? 01010000 = 11001000

RC4 Decryption
11001000 ? 01010000 = 10011000
```

CAST Algorithm

CAST is a general procedure for creating a family of block ciphers. Individual ciphers have names like CAST-128 and CAST-256.

The CAST algorithm is generally used in IP security and follows the Fiestal structure, as the Fiestal structure divides the plain text into equal halves and does the encryption process.

They use large S-boxes, 8*32 rather than the 6*4 of DES.

They are designed for software implementation.

The CAST S-boxes use bent functions as their columns.

S-boxes meet the avalanche criterion, which means that every bit of input and every bit of round key affect every bit of round output and complement any input bit has exactly a 50% chance of changing any given output bit.

CAST-128

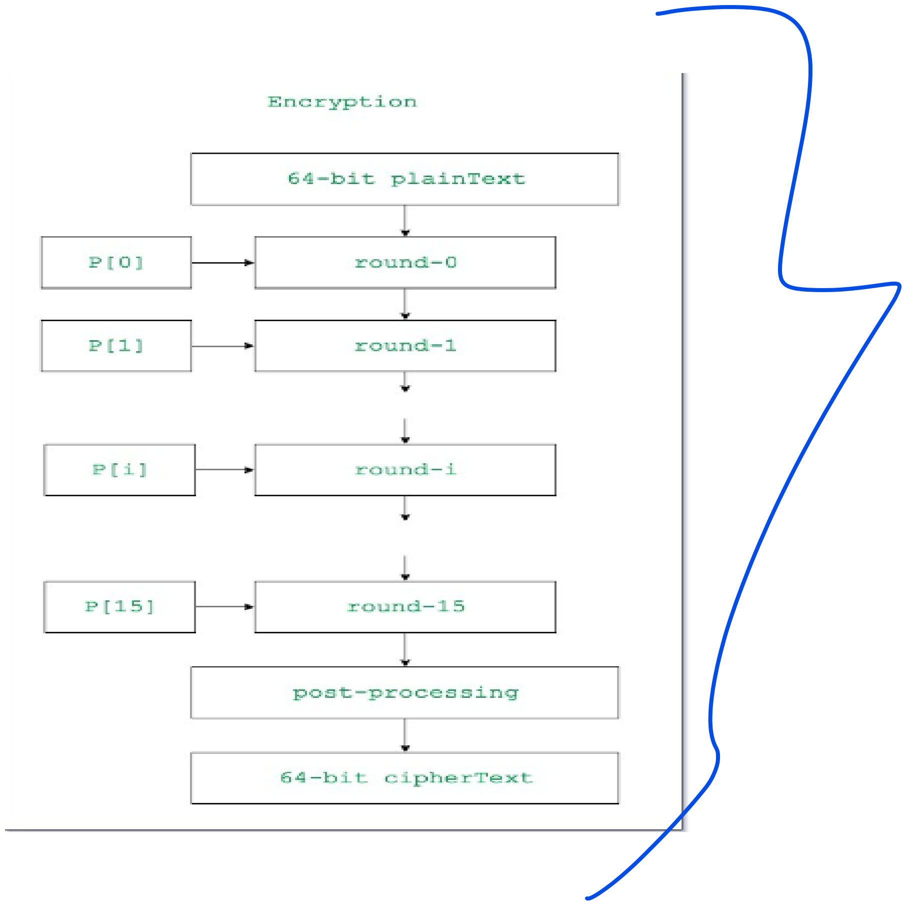
CAST-128 also known as CAST5 is the most widely used CAST Cipher. It is a symmetric key block cipher that is used in a number of products as the default cipher in some versions of GPG and PGF.

- It is a Feistel Cipher with 64-bit blocks and 16 rounds.
- The key size varies from 40 bits to 128 bits.
- There are eight 8*32 S-boxes. Out of these eight boxes, four are used in key scheduling and rest four boxes are used in actual encryption.
- Round keys are 37 bits.
- The F function XORs the input with 32 bits of round key, breaks the result into bytes, and runs each byte through a different S-box to get four 32-bit results.
- Those are combined nonlinearly using different combining functions in different rounds.
- The output is given a rotation that is controlled by other 5 round key bits.

Blowfish Algorithm

Blowfish is an encryption technique designed by Bruce Schneier in 1993 as an alternative to DES Encryption Technique. It is significantly faster than DES and provides a good encryption rate with no effective cryptanalysis technique found to date. It is one of the first, secure block cyphers not subject to any patents and hence freely available for anyone to use. It is symmetric block cipher algorithm.

- blockSize: 64-bits
- keySize: 32-bits to 448-bits variable size
- number of subkeys: 18 [P-array]
- number of rounds: 16
- number of substitution boxes: 4 [each having 512 entries of 32-bits each]



Linear Cryptanalysis

Linear cryptanalysis is a strong cryptanalytic tool regarding cryptanalysis of block ciphers. When using linear cryptanalysis, an adversary attempt to discover a linear expression that approximates a non-linear function with a probability different than 1/2.

When a best approximation, it includes a relation between the plaintext and ciphertext, is discovered, the adversary gains information about the secret key. The approximation has the form –

$$P_i \oplus .. \oplus P_j \oplus C_k \oplus C_1 = k_m \oplus k_n$$

with P_i … P_j being plaintext bits, C_k … C_l ciphertext bits and K_m … K_n key bits. The approximation influence with some probability p, and its quality is generally computed by the bias which is defined as $\epsilon = \left|p - \frac{1}{2}\right|$.

Differential Cryptanalysis

Differential cryptanalysis is a common style of cryptanalysis relevant frequently to block ciphers, but it can also to stream ciphers and cryptographic hash functions. In the generous sense, it is the study of how differences in information input can influence the resultant difference at the output.

In the case of a block cipher, it defines a group of techniques for tracing differences through the web of transformation, finding where the cipher exhibits non-random behaviour and exploiting such properties to find the secret key