2. Given:

$n = 17$

$a = 5$

Private key of Alice = 4

Private key of Bob = 6

Public key of Alice:

$= 5^{\text{privated key of Alice}} \bmod 17$

$= 5^4 \bmod 17$

$= 13$

Public key of Bob:

$= 5^{\text{private key of Bob}} \bmod 17$

$= 5^6 \bmod 17$

$= 2$

Secret key obtained by Alice:

$= 2^{\text{private key of Alice}} \bmod 17$

$= 2^4 \bmod 17$

$= 16$

Secret key obtained by Bob:

$= 13^{\text{private key of Bob}} \bmod 17$

$= 13^6 \bmod 17$

$= 16$

Finally both parties get same value of secret key.

$\therefore$ Value = 16

$\therefore$ Option (A) : 16 is correct.

Q4.

String = "GEEKSFORGEEKS"
keyword = "SHARAN"

```
def generateKey(string, key):
    key = list(key)
    if len(string) == len(key):
        return(key)
    else:
        for i in range(len(string) - len(key)):
            key.append(key[i % len(key)])
    return("".join(key))

def encrypt_cipherText(string, key):
    cipher_text = []
    for i in range(len(string)):
        x = ((ord(string[i]) + ord(key[i])) % 26) + ord('A')
        cipher_text.append(chr(x))
    return("".join(cipher_text))

key = generatekey(string, keyword)

print('Original Message:', string)
print('Keyword:', keyword)
cipher_text = encrypt_ciphertext(string, key)
print('Ciphertext:', cipher_text)
```

Output: Original Message : GEEKSFORGEEKS
Keyword : SHARAN
Ciphertext : YLEBSSGYGVEXK

Decryption of Vignere Cipher:

```
Ciphertext = "YLEBSSGYGVEXK"
Keyword = "SHARAN"
def generateKey (ciphertext, key):
    Key = list (key)
    if len (string) == len (key):
        return key
    else:
        for i in range (len (string) - len (key)):
            key. append (key [i % len (key)])
        return ("".join (key))

def decrypt_originaltext (ciphertext, key):
    origtext = []
    for i in range (len (ciphertext)):
        x = ((ord (ciphertext [i] - ord (key [i])) % 26) + ord ('A')
        origtext. append (chr (x))
    return ("".join (origtext))

Key = generateKey (ciphertext, keyword)
print ('Ciphertext:', ciphertext)
print ('Keyword:', keyword)
string = decrypt_originaltext (ciphertext, key)
print ('Original text=', string)
```

OUTPUT:

```
ciphertext = 'YLEBSSGYGVEXK'
Keyword = 'SHARAN'
original text = 'GEEKSFORGEEKS'
```