b $q_p = 17$ $q_B = 5$

Alice Secret key $a = 4$ Bob secret key (b) $= 6$

Public key of alice $= q^a \bmod P$

(A) $= 5^4 \bmod 17$

$= 13$

Public key of Bob $= q^b \bmod P$

(B) $= 5^6 \bmod 17$

$= 2$.

Secret key obtained by alice = Public key of bob$^a \bmod P$

$= 2^4 \bmod 17$

$= 16$

Secret key obtained by bob = Public key of alice$^b \bmod P$

$= 13^6 \bmod 17$

$= 16$ //

Ans is a. 16 //

Question (4)

Encryption: string = "Aaron Mendonca"
Keyword = "HELLO"

```
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_cipher(string, key):
    cipher_text = []
    for i in range(len(string)):
        x = (ord(string[i]) + ord(key[i]) %26)
        ciphertext.append(chr(x))
        return("".join(ciphertext))

key = generateKey(string, keyword)
cipher_text = encrypt_cipher(string, key)

print("Plain text", string)
print("Keyword:", keyword)
print("Cipher", ciphertext)
```

```
string = "GEEKSFORGEEKS"
keyword = "AYUSH"
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 originaltext(ciphertext, key):
    orig_text = []
    for i in range(len(ciphertext)):
        x = (ord(ciphertext[i]) - ord(key[i]) + 26) % 26
        x += ord('A')
        orig_text.append(chr(x))
    return("".join(orig_text))
key = generatekey(string, keyword)
print("Original text:", originaltext(string, key))
```