Name - Yash Lakhtariya
Enrollment number - 21162101012
Branch - CBA Batch - 61
INS Practical 8

Aim : For encryption purpose two parties Alice and Bob want to share some secrete key over a communication network, Which Key Exchange algorithm is best suited for this scenario. Prepare suitable environment for the same.

Code :

```
import random
import YSL_io as ysl

p = int(ysl.inputORNG("\nEnter any prime number: "))
alpha = []
l1 = []


def check(a, b):
    for i in range(1, b):
        if i in a:
            continue
        else:
            return False


for i in range(2, p):
    for j in range(1, p):
        val = (i**j) % p
        l1.append(val)
    alpha.append(l1)
    l1 = []

fin_alpha = []
for i in range(len(alpha)):
    if check(alpha[i], p) != False:
        fin_alpha.append(alpha.index(alpha[i]) + 2)
```

Name - Yash Lakhtariya
Enrollment number - 21162101012
Branch - CBA Batch - 61
INS Practical 8

```
ysl.printGRN("The available values for alpha will be: ", end="")
print(fin_alpha)

a = random.randint(1, p)
b = random.randint(1, p)
while a == b:
   b = random.randint(1, p)

sel_alp = min(fin_alpha)
if sel_alp == 2:
   fin_alpha.remove(2)
   sel_alp = min(fin_alpha)

public_A = (sel_alp**a) % p
public_B = (sel_alp**b) % p
c = a * b

key_a = (sel_alp**c) % p
key_b = (sel_alp**c) % p

ysl.printMGNTA(f"\nSelected value for alpha: {sel_alp}")
ysl.printBLU(f"\nPublic_A: {public_A}")
ysl.printBLU(f"Public_B: {public_B}")
ysl.printGRN(f"\nSelected key of Sender Side : {key_a}")
ysl.printGRN(f"Selected key of Receiver Side : {key_b}")

if key_a == key_b:
   ysl.printRED("\nKey Matched. Exchange of key was successful")
else:
   ysl.printRED("\nKey Not Matched. Exchange of key was unsuccessful")
```

Name - Yash Lakhtariya
Enrollment number - 21162101012
Branch - CBA Batch - 61
INS Practical 8

Output :