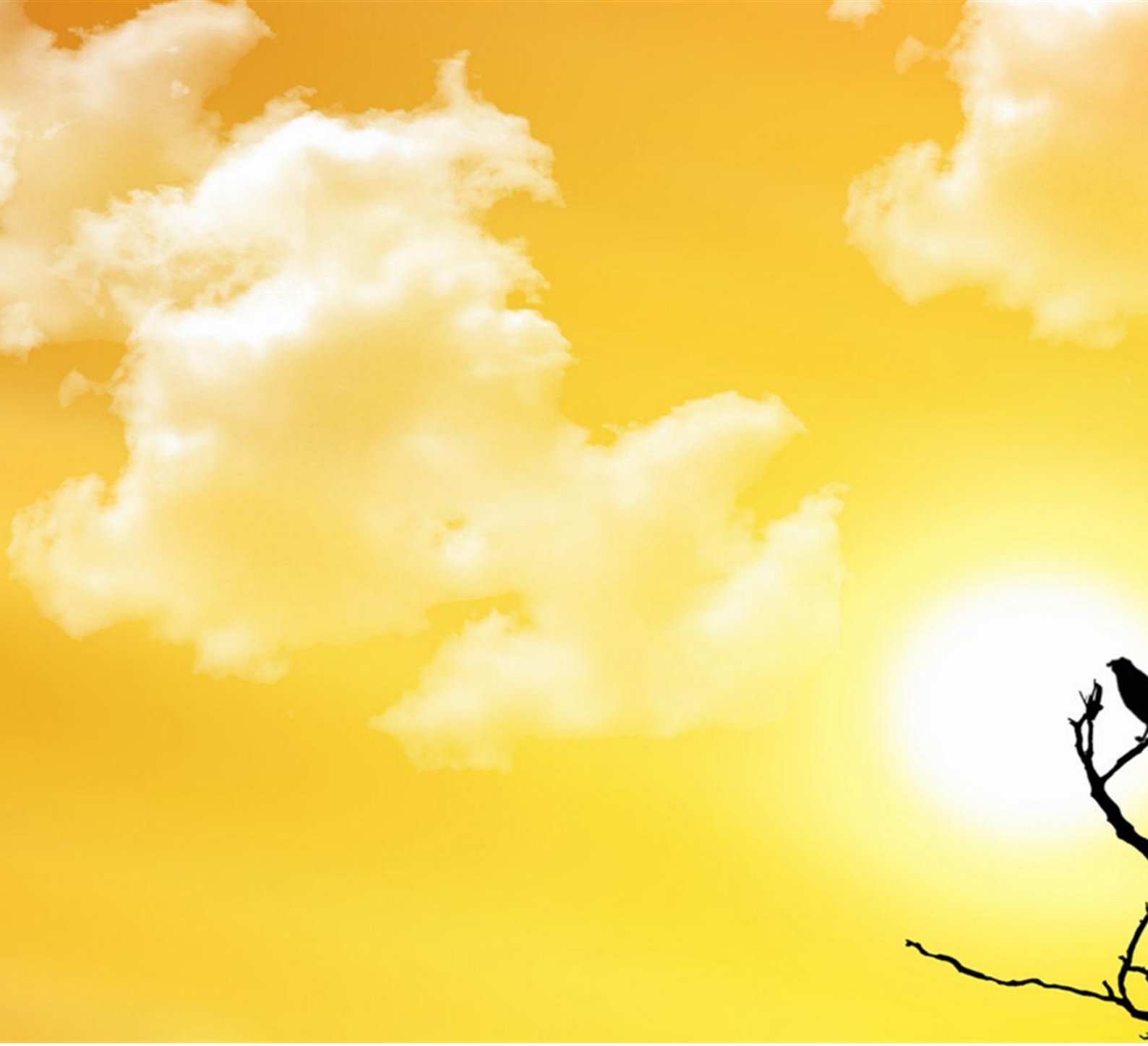


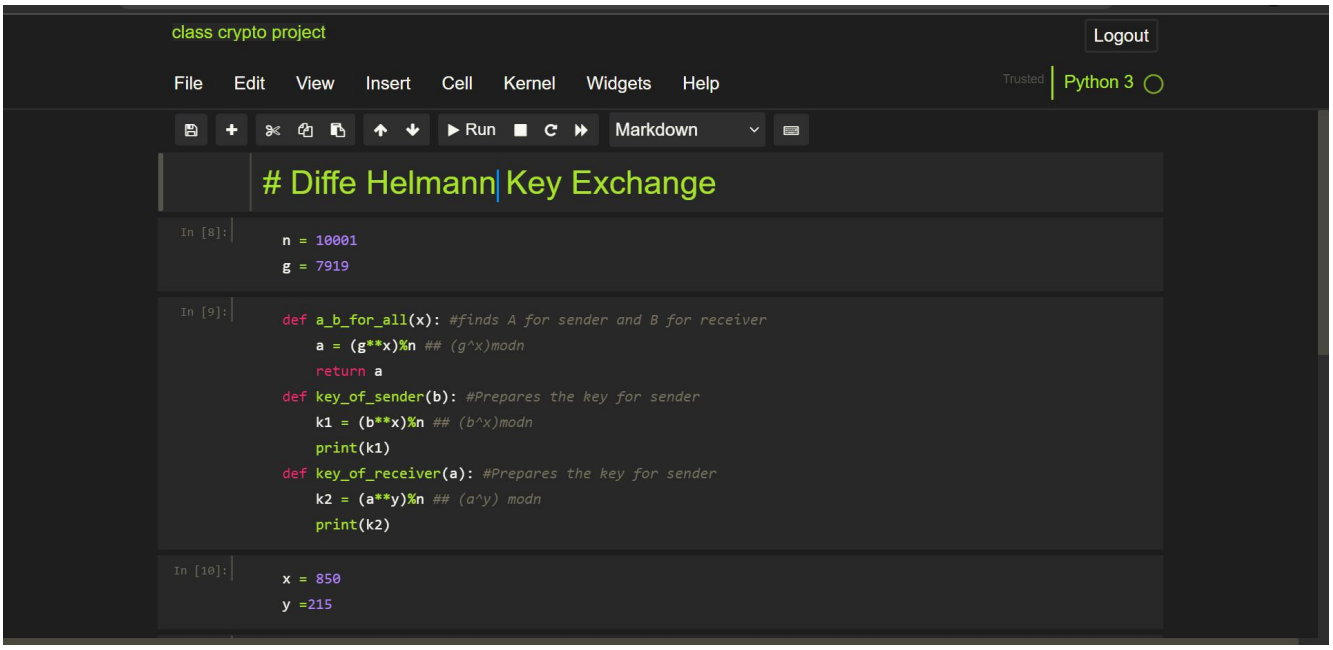
Diffie Hellman Key exchange

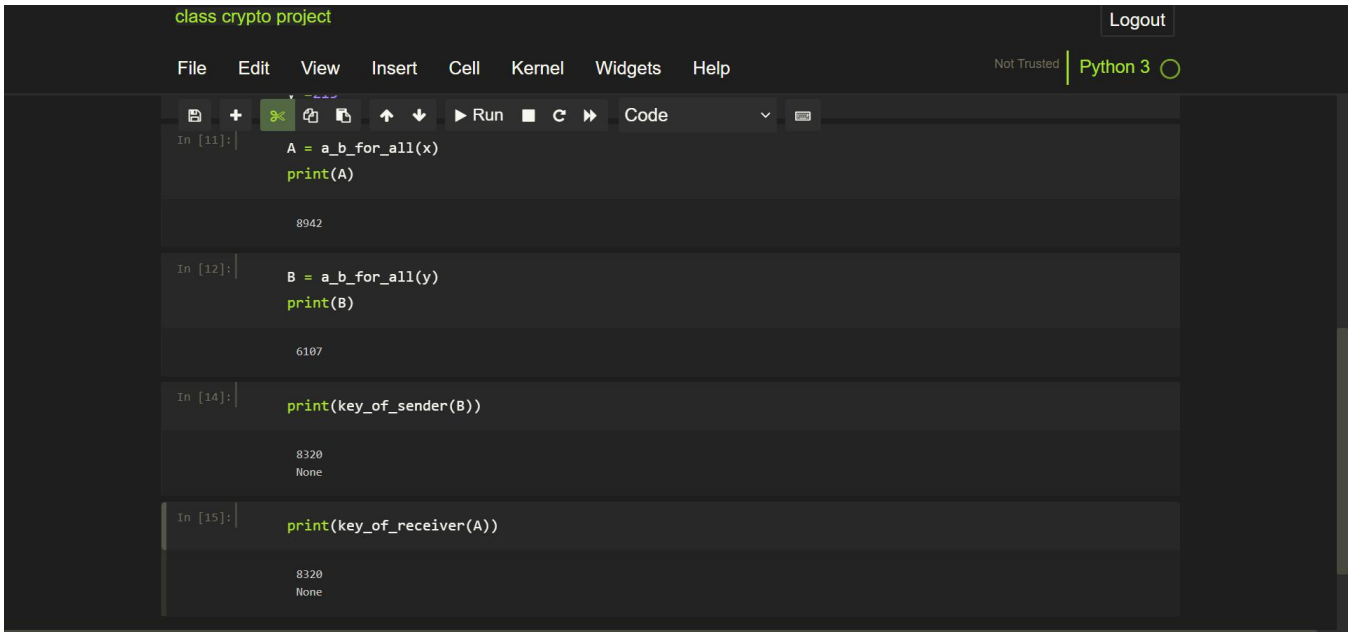
Arpan Maity	1905850
Srijita Guha Roy	1905849
Sucheta Bhattacharjee	1906205
Ritobina Ghosh	1905043
Harshit Kumar	1905852
Osama Mahmoud	1905529

Code (Python 3)

```
n = 10001
g = 7919

def a_b_for_all(x): #finds A for sender and B for receiver
    a = (g**x)%n ## (g^x)modn
    return a
def key_of_sender(b): #Prepares the key for sender
    k1 = (b**x)%n ## (b^x)modn
    print(k1)
def key_of_receiver(a): #Prepares the key for sender
    k2 = (a**y)%n ## (a^y) modn
   print(k2)

x = 850
y =215

A = a_b_for_all(x)
print(A)

B = a_b_for_all(y)
print(B)

print(key_of_sender(B))

print(key_of_receiver(A))
```

Screenshots





Explanation:

The process of the code is explained below in steps

- In Diffe Helmann key exchange algorithm , n and g are chosen as large prime numbers. For sake of coding, we chose n and g as medium sized prime numbers, where n = 10001 and g = 7919
- x and y are secret numbers of sender and receiver respectively. For this code we chose x = 850 and y = 215
- According to Diffe Helmann key exchange A = (g^x)modn (for sender) and B = (g^y)modn (for receiver). In our code we wrote a function named a_b_for_all which returned A to receiver and B to sender
- We defined function called key_of_sender, which returns sender its key, using the formula Key = (B^x)modn
- And we defined function called key_of_receiver, which returns receiver its key, using the formula Key = (A^x)modn
- After computation, both the keys must be equal, and according to our code, the value of our key is 8320
- The screenshots of page 3 can be referred for the details of our code along with its output