**Name:Hafiz Yahya Iqbal Reg#16pwcse1464**

**Qno1:** Implement the Diffie-Hellman protocol in Matlab. It should pick the prime numbers randomly for both users A and B and be able to calculate the common value.

disp('DF Protocol');

prompt = 'Secret Nummber for a: ';

a = input(prompt)

prompt = 'Secret Nummber for b: ';

b = input(prompt)

num = 2:50;

qm = num(isprime(num));

p = qm(randi(numel(qm)))

qm = num(isprime(num));

g = qm(randi(numel(qm)))

GA = power(g, a);

A = mod(GA, p);

disp(' Alice secret number : ');

disp(A);

GB = power(g, b);

B = mod(GB, p);

disp(' Bob secret number : ');

disp(B);

SA = power(B, a);

s\_A = mod(SA , p);

disp(' Alice computes: ');

disp(s\_A);

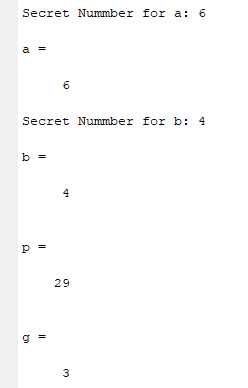
SB = power(A, b);

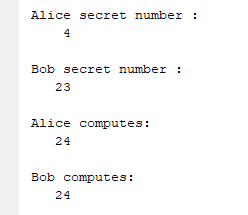
s\_B= mod(SB, p);

disp(' Bob computes: ');

disp(s\_B);

**Output**





**Qno2:** Implement the RSA protocol in Matlab.

disp('RSA Protocol');

num = 2:30;

qm = num(isprime(num));

p = qm(randi(numel(qm)))

qm = num(isprime(num));

q = qm(randi(numel(qm)))

n = p \* q

np = (p - 1)\*(q - 1)

num1 = 2 : np;

qm = num1(isprime(num1));

e = prime1(randi(numel(qm)))

[dev,z1,z2] = gcd(e,np);

if dev==1

d = mod(z1,np);

disp('value of d : ');

disp(d);

end

prompt = 'Enter the value of m: ';

m = input(prompt);

m\_power = power(m, e);

m\_mod= mod(m\_power, n);

disp(' Encryption: ');

disp(m\_mod)

**Output**

