Exp 2b :Diffie-Hellman Key Exchange Algorithm

Code:

#include <math.h>

#include <stdio.h>

long long int power(long long int a, long long int b,

long long int P)

{

if (b == 1)

return a;

else

return (((long longint)pow(a, b)) % P);

}

int main()

{

long long int P, G, x, a, y, b, ka, kb;

printf("Enter the prime number P: ");

scanf("%lld", &P);

printf("Enter the primitive root G: ");

scanf("%lld", &G);

printf("Enter the private key a for Jeff: ");

scanf("%lld", &a);

x = power(G, a, P);

printf("Enter the private key b for Rose: ");

scanf("%lld", &b);

y = power(G, b, P);

ka = power(y, a, P);

kb = power(x, b, P);

printf("The value of P : %lld\n", P);

printf("The value of G : %lld\n\n", G);

printf("The private key a for Jeff : %lld\n", a);

printf("The private key b for Rose : %lld\n\n", b);

printf("Secret key for the Jeff is : %lld\n", ka);

printf("Secret Key for the Rose is : %lld\n", kb);

return 0;

}

Output:

