#include <stdio.h>

#include <math.h>

long long power(long long base, long long exp, long long mod) {

long long result = 1;

base = base % mod;

while (exp > 0) {

if (exp % 2 == 1)

result = (result \* base) % mod;

exp = exp >> 1;

base = (base \* base) % mod;

}

return result;

}

int main() {

long long P, G, a, b;

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

scanf("%lld", &P);

printf("Enter a primitive root modulo P (G): ");

scanf("%lld", &G);

printf("Enter private key of User A: ");

scanf("%lld", &a);

printf("Enter private key of User B: ");

scanf("%lld", &b);

long long A = power(G, a, P);

long long B = power(G, b, P);

long long secretA = power(B, a, P); t

long long secretB = power(A, b, P);

printf("\nPublic key of User A: %lld", A);

printf("\nPublic key of User B: %lld", B);

printf("\n\nShared secret computed by User A: %lld", secretA);

printf("\nShared secret computed by User B: %lld\n", secretB);

if (secretA == secretB)

printf("\nKey exchange successful. Shared secret established.\n");

else

printf("\nKey exchange failed.\n");

return 0;

}

A screenshot of a computer program

AI-generated content may be incorrect.