```c
// Write a program to implement the Diffie-Hellman Key Exchange algorithm.

#include <stdio.h>
 // Function to compute `a^m mod n`
int compute(int a, int m, int n)
{   int y = 1;
    while (m > 0)
    {  // fast exponention
        if (m % 2 == 1)
            y = (y*a) % n;
        a = a*a % n;
        m /= 2;
    }
    return y;
}
int main()
{   int p,g;//p=prime number,g=alpha
    printf("Enter a prime number and alpha value(alpha should satisfy the condition)");
    scanf("%d%d",&p,&g);
    int a, b;    // `a` â€“ A's secret key, `b` â€“ B's secret key.
    int A, B;    // `A` â€“ A's public key, `B` â€“ B's public key
    // choose a secret integer for A's private key (only known to A)
    a = rand();
    printf("Xa=%d\n",a);
    // Calculate A's public key (A will send `A` to B)
    A = compute(g, a, p);
    printf("Ya=%d\n",A);
    // choose a secret integer for B's private key (only known to B)
    b = rand();
    printf("Xb=%d\n",b);
    // Calculate B's public key (B will send `B` to A)
    B = compute(g, b, p);
    printf("Yb=%d\n",B);
    // A and B Exchange their public key `A` and `B` with each other

    // Find secret key
    int keyA = compute(B, a, p);
    int keyB = compute(A, b, p);
    printf("A's secret key is %d\nB's secret key is %d\n", keyA, keyB);
}
```