EX-NO-10-Diffie-Hellman-Key-Exchange-Algorithm

AIM:

To Implement Diffie Hellman Key Exchange Algorithm

Algorithm:

1. Diffie-Hellman Key Exchange is used for securely sharing a secret key between two parties over an insecure channel.
2. Initialization: Agree on a large prime number (p) and a primitive root (g) modulo (p) (both are public values).
3. Key Exchange Process:
   - Each party selects a private key and calculates their public key using the formula (g^{\text{private key}} \mod p).
   - Each party then shares their public key with the other.
4. Secret Key Computation:
   - Each party computes the shared secret key using the received public key and their own private key.
5. Security: The difficulty of computing discrete logarithms ensures that the shared key remains secure even if public values are intercepted.

Program:

```
Register No: 212223100035
Name: Muhammad Afshan A
```

```
#include <stdio.h>
long long int mod_exp(long long int base, long long int exp, long long int mod) {
    long long int result = 1;
    while (exp > 0) {
        // If exp is odd, multiply base with result
        if (exp % 2 == 1)
            result = (result * base) % mod;

        // Now exp must be even, so divide by 2 and square the base
        exp = exp >> 1; // equivalent to exp = exp / 2
        base = (base * base) % mod;
    }
    return result;
}

int main() {
    long long int P, G, a, b, x, y, ka, kb;

    printf("Enter the value of P: ");
    scanf("%lld", &P);
```

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

    printf("Enter the value of G (Primitive root of P): ");
    scanf("%lld", &G);
    printf("The value of G: %lld\n\n", G);

    printf("Enter the private key for Muhammad (a): ");
    scanf("%lld", &a);
    x = mod_exp(G, a, P);


    printf("Enter the private key for Afshan (b): ");
    scanf("%lld", &b);
    y = mod_exp(G, b, P);


    ka = mod_exp(y, a, P); // Alice computes the shared key ka = y^a % P
    kb = mod_exp(x, b, P); // Bob computes the shared key kb = x^b % P

    printf("\nShared secret key for Alice : %lld\n", ka);
    printf("Shared secret key for Bob : %lld\n", kb);

    return 0;
}
```

Output:

Output Clear

```
Enter the value of P: 6
The value of P: 6
Enter the value of G (Primitive root of P): 14
The value of G: 14

Enter the private key for Muhammad (a): 18
Enter the private key for Afshan (b): 21

Shared secret key for Alice : 4
Shared secret key for Bob : 4


=== Code Execution Successful ===
```

Result:

The program is executed successfully