EX NO: **DSA** REG NO:210701290

DATE:

AIM:-

To implement Digital Signature Algorithm[DSA] using Java.

ALGORITHM:-

STEP 1:Get the prime number p and its divisor q from the user.

STEP 2:Get the value of h from the user.

STEP 3:Compute the value of g.

STEP 4:Get the private key xa from the user.

STEP 5:Compute the user's public key y.

STEP 6:Get the per-message secret key k and hash value of message M.

STEP 7:Compute the value of z using g, k & p.

STEP 8:Compute z % q to get the value of r.

STEP 9:Compute the multiplicative inverse.

STEP 10:Compute the value of s.

STEP 11:Print the signature (r, s).

PROGRAM:-

```
import java.util.Scanner;
public class Main {
    public static int power(int x, int y, int p) {
        int res = 1;
        x = x % p;
```

```
    while (y > 0) {
        if (y % 2 == 1)
            res = (res * x) % p;
        y = y >> 1;
        x = (x * x) % p;
    }
    return res;
}
public static int multiplicativeInverse(int a, int b, int n) {
    int sum, x, y;
    for (y = 0; y < n; y++) {
        for (x = 0; x < n; x++) {
            sum = a * x + b * (-y);
            if (sum == 1)
                return x;
        }
    }
    return -1; // Return -1 if inverse doesn't exist
}
public static void main(String[] args) {
    Scanner scanner = new Scanner(System.in);
    int p, q, h, g, r, s, t, x, y, z, k, inv, hash;
```

```
        System.out.println("Enter prime number p and prime divisor q of (p-1): ");
        p = scanner.nextInt();
        q = scanner.nextInt();
        System.out.println("Enter h such that it's greater than 1 and less than (p-1):");
        h = scanner.nextInt();
        t = (p - 1) / q;
        g = power(h, t, p);
        System.out.println("Enter user's private key (0 < x < q):");
        x = scanner.nextInt();
        y = power(g, x, p);
        System.out.println("Enter user's per-message secret key (0 < k < q):");
        k = scanner.nextInt();
        System.out.println("Enter the hash(M) value:");
        hash = scanner.nextInt();
        z = power(g, k, p);
        r = z % q;
        inv = multiplicativeInverse(k, q, p);
        s = (inv * (hash + x * r)) % q;
        System.out.println("\n*********Computed Values*********");
        System.out.println("g = " + g);
        System.out.println("y = " + y);
        System.out.println("Generated Signature Sender = (" + r + ", " + s + ")");
    }
```

}

OUTPUT:-

```
Enter prime number p and prime divisor q of (p-1):
1279
71
Enter h such that it's greater than 1 and less than (p-1):
3
Enter user's private key (0 < x < q):
15
Enter user's per-message secret key (0 < k < q):
10
Enter the hash(M) value:
123

*********Computed Values*********
g = 1157
y = 851
Generated Signature Sender = (32, 39)
```

RESULT:-