PANIMALAR INSTITUTE OF TECHNOLOGY

Department of Computer Science and Engineering

Academic Year: 2019-2020 (Odd Semester)

INNOVATIVE TOOLS

Degree, Semester& Branch: VII Semester B.E. Computer Science and Engineering
Course Code & Title: CS6701 Cryptography and Network Security
Name of the Faculty member: Dr.V.Subedha, Mrs.S.Hemamalini & Mrs.S.Annie Sheryl

TOPIC: DSA Algorithm **Date: 30.08.2019**

Group Activity: Respond, React and Reply

Group Activity:

Working in small groups provides learners with opportunities to articulate ideas and understandings, uncover assumptions and misconceptions, and negotiate with others to create products or reach consensus.

When Should Group Activity Be Used?
This is a great activity for online classrooms. If a student is delayed in responding/reacting/replying, the instructor can give "behind the scene" nudges.

Outcome:

- Ensures active participation of students
- Enables students to improve their comprehensive, interpersonal skills.

Suggested Activity: Respond/React/Reply

- ✓ Break students up into small groups.
- ✓ Provide students with a prompt. The prompt can be a targeted question, written passage/text, or argument.
- ✓ Each student then responds to the prompt on their own in writing. After each student has had a chance to write their response, have them read and share their response with the group.
- ✓ Each student then reacts to each of the other group members' responses.
- ✓ Then, the student replies to each of the reactions to their own response.

Evaluating Digital Signature Algorithm (DSA):

The sample test case worked out by Team A, B and C could be as follows:

- ❖ Team A is given with a smaller prime divisor q=3 and prime modulus p=7.

The process of generating the public key and private key by Team A is as follows:

```
q = 3       # selected prime divisor
p = 7       # computed prime modulus: (p-1) mod q = 0
g = 4       # computed: 1 < g < p, g**q mod p = 1
            #           and g = h**((p-1)/q) mod p
            #           4**3 mod 7 = 1: 64 mod 7 = 1
x = 5       # selected: 0 < x < q
y = 2       # computed: y = g**x mod p = 4**5 mod 7
{7,3,4,2}   # the public key: {p,q,g,y}
{7,3,4,5}   # the private key: {p,q,g,x}
```

- With the private key {p,q,g,x}={7,3,4,5}, the process of generating a digital signature out a message hash value of h=3 by Team B can be illustrated as:

```
h = 3       # the hash value as the message digest
k = 2       # selected: 0 < k < q
r = 2       # computed: r = (g**k mod p) mod q = (4**2 mod 7) mod 3
i = 5       # computed: k*i mod q = 1: 2*i mod 3 = 1
s = 2       # computed: s = i*(h+r*x) mod q = 5*(3+2*5) mod 3
{2,2}       # the digital signature
```

- The process of verifying the digital signature {r,s}={2,2} with the public key {p,q,g,y}={7,3,4,2} by Team C can be illustrated as:

```
h = 3       # the hash value as the message digest
w = 5       # computed: s*w mod q = 1: 2*w mod 3 = 1
u1 = 0      # computed: u1 = h*w mod q = 3*5 mod 3 = 0
u2 = 1      # computed: u2 = r*w mod q = 2*5 mod 3 = 1
v = 2       # computed: v = (((g**u1)*(y**u2)) mod p) mod q
            #             = (((4**0)*(2**1)) mod 7) mod 3 = 2
v == r      # verification passed
```