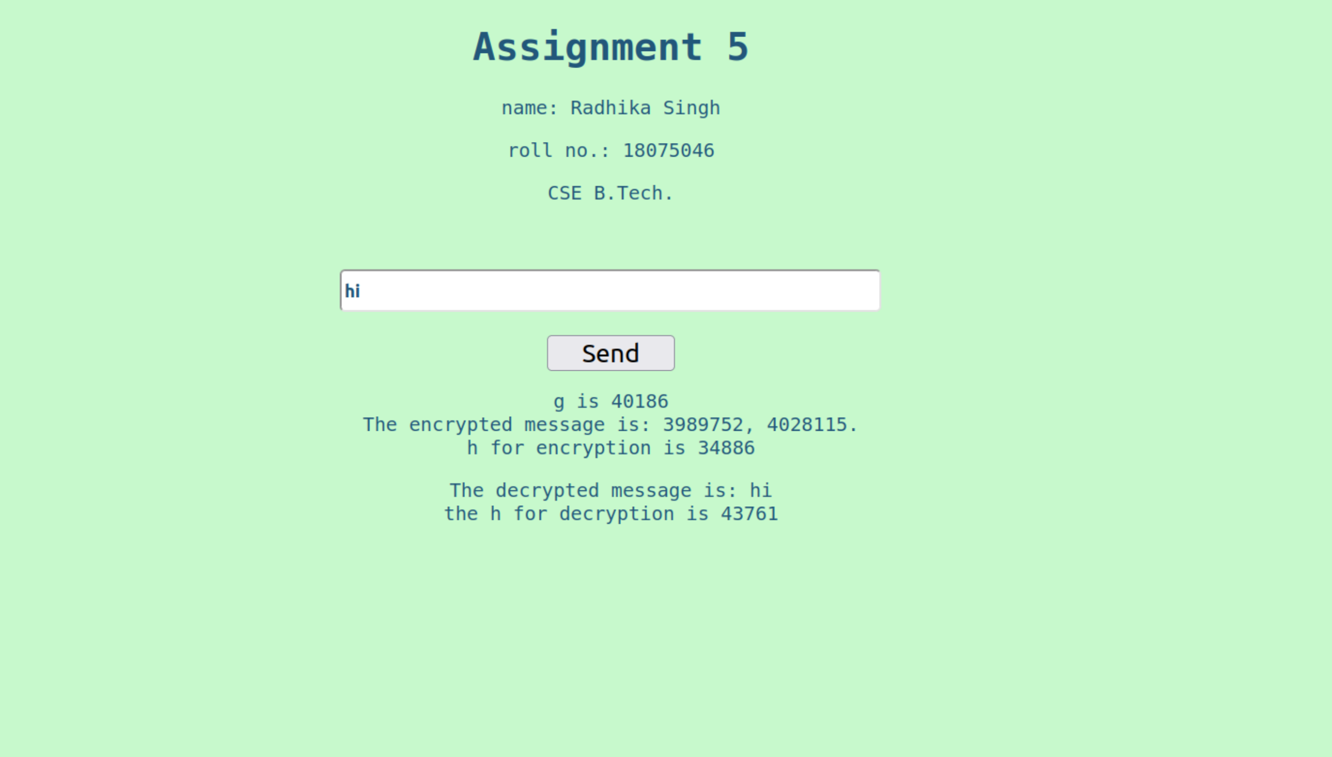
Name : Radhika Singh
Roll No. : 18075046
Dept. : CSE

Assignment 5

Screenshots:



Assignment 5

name: Radhika Singh

roll no.: 18075046

CSE B.Tech.

hi

```
g is 40186
The encrypted message is: 3989752, 4028115.
h for encryption is 34886

The decrypted message is: hi
the h for decryption is 43761
```

Source code:

```
<!DOCTYPE html>
<html lang="en" dir="ltr">
  <head>
          <meta charset="utf-8">
          <title>El Gammal</title>
          <style media="screen">
```

```
    body{
    background-color: #C7F9CC;
    color: #22577A;
    font-family: monospace;
    font-size: 1.2em;

    }

    .container{
    margin-left: 5em;
    margin-right: 5em;
    margin-top: auto;
    margin-bottom: auto;
    }

    input{
    height: 2em;
    width: 30em;
    border-radius: 0.3em;
    color: inherit;
    font-weight: bold;
    }
    .center{
    text-align: center;
    }

    button{
    width: 5em;
    height: 1.4em;
    font-size: 1em;
    }
    table{
    width: 100%;
    }
    td{
    padding: 5%;
    width: 50%;
    }

    </style>
</head>
<body>
    <div class="container center">
    <h1>Assignment 5</h1>
```

```
  }




          <p>name: Radhika Singh</p>
          <p>roll no.: 18075046</p>
          <p>CSE B.Tech.</p>





          <br>
          <br>
          <input type="text" name="" value="" id="usermsg">
          <br>
          <br>
          <button type="button" name="send"id="usersend">Send</button>
          <br>
          <p id="userencrec">The encrypted message is: </p>
          <p id='userdecrec'>The decrypted message is: </p>

          </div>

<script type="text/javascript">
  // user 1 variables
  var userMsg = document.getElementById("usermsg");
  var userSend = document.getElementById("usersend");
  var userEncRec = document.getElementById("userencrec");
  var userDecRec = document.getElementById("userdecrec");

  var user1PrivateKey;
  var user1PublicKey;

  // user 2 variables
  var user2PrivateKey;
  var user2PublicKey;

  // function to compute gcf of two numbers
  const gcd = function(a, b){
          if(a<b){
          return gcd(b,a);
          }
          if(a%b === 0){
          return b;
          }
          return gcd(b,a%b);
  }
```

```
const genKey = function(q){

        return Math.trunc(key);
}
// function to generate key
const generateKey = function(q){
        var key = Math.random()*(q-Math.pow(2,6));
        key+=Math.pow(2,6);
        key = Math.trunc(key);
        while(gcd(q,key)!=1){
        key=Math.random()*(q-Math.pow(2,6));
        key+=Math.pow(2,6);
        key = Math.trunc(key);
        }
        return key;
}

// modular exponentiation
const power = function(a,b,c){
        var x = 1;
        var y = Math.trunc(a);
        while(b>0){
        if(b%2!=0){
        x=(x*y)%c;
        }
        y=(y*y)%c;
        b=Math.trunc(b/2);
        }
        return x%c;
}

// encryption function
const encrypt = function(msg,q,h,g){
        var encMsg = [];
        var k = user1PrivateKey;

        s = power(h,k,q);
        p = power(g,k,q);

        for(var i=0;i<msg.length;i++){
        encMsg.push(s*msg.charCodeAt(i));
        }
        return encMsg;
}
```

```
// decryption function
const decrypt = function(encMsg,p,key,q){
        decMsg = "";
        var h = power(p,key,q);
        for(var i=0;i<encMsg.length;i++){
        decMsg+=String.fromCharCode(Math.trunc(encMsg[i]/h));
        }
        return decMsg;
}


var primes = [];
var flag = true;
for(var i=Math.pow(2,6)+1;i<Math.pow(2,16);i+=2){
        flag =true;
        for(j=2;j<i;j++){
        if(i%j==0){
        flag = false;
        break;
        }
        }
        if(flag){

        primes.push(i);
        }
}


// generate q
var q = primes[Math.trunc(Math.random()*primes.length)];

// generate g
var g = Math.trunc((Math.random()*(q-2))+2);
// private key of user 2
user2PrivateKey = generateKey(q);
// h of user 2
var h = power(g,user2PrivateKey,q);
// public key of user 2
user2PublicKey = {'g':g,'h':h,'q':q};
// private key of user 1
user1PrivateKey = generateKey(q);
// public key of user 1
user1PublicKey = {'g':g,'h':power(g,user1PrivateKey,q),'q':q};
```

```
  // var encMsg = encrypt(msg,q,user2PublicKey['h'],g,1);
  // var decMsg = decrypt(encMsg['encMsg'],user1PublicKey['h'],user2PrivateKey,q);
  // console.log(decMsg);

  userSend.addEventListener('click',()=>{
          var msg = userMsg.value;
          var encMsg = encrypt(msg,q,user2PublicKey['h'],g,1);
          var decMsg = decrypt(encMsg,user1PublicKey['h'],user2PrivateKey,q);
          userEncRec.innerText = "g is "+g+"\nThe encrypted message is: "+encMsg.join(', ')+".\n
h for encryption is "+user2PublicKey['h'];
          userDecRec.innerText = "The decrypted message is: "+decMsg+"\n the h for decryption
is "+user1PublicKey['h'];
  });

</script>
  </body>
</html>
```

Github link: https://github.com/Radhika-singh/Assignment5