Q1

```
function r = recursie_factorial(n)
    if(n == 0)
        r = 1;
    else
        r = n*recursie_factorial(n-1);
    end
end

n = input("Find factorial of? ")
```

```
n =
5
```

```
fprintf("Factorial using own fuction: %d",recursie_factorial(n))
```

```
Factorial using own fuction: 120
```

```
fprintf("Factorial using factprial() fuction: %d",factorial(n))
```

```
Factorial using factprial() fuction: 120
```

Q2

```
function r = my_gcd(a,b)
    if(b ~= 0)
        r = my_gcd(b,mod(a,b));
    else
        r = a;
    end
end

function r = my_lcm(a,b)
    r = a * b / my_gcd(a,b);
end

for i = 0:4
    a = input("Enter 1st number: ")
    b = input("Enter 2nd number: ")
    fprintf("HCF OF NUMBERS IS %d, LCM IS %d",my_gcd(a,b),my_lcm(a,b))
    fprintf("LCM using lcm() is %d",lcm(a,b))
end
```

```
a =
6
b =
36
HCF OF NUMBERS IS 6, LCM IS 36
LCM using lcm() is 36
a =
4
b =
14
HCF OF NUMBERS IS 2, LCM IS 28
LCM using lcm() is 28
a =
5
```

```
b =
46
HCF OF NUMBERS IS 1, LCM IS 230
LCM using lcm() is 230
a =
7
b =
25
HCF OF NUMBERS IS 1, LCM IS 175
LCM using lcm() is 175
```

Q3

```
function r = my_pow(a,n)
    if(n == 0)
        r = 1;
    else
        r = a*my_pow(a,n-1);
    end
end

for i = 0:2
    a = input("Enter number: ")
    n = input("Enter its exponent: ")
    fprintf("a^n using recursive function is %d",my_pow(a,n))
    fprintf("difference between recursive function and pow() is %d",my_pow(a,n)-
power(a,n))
end
```

```
a =
2
n =
100
a^n using recursive function is 1.267651e+30
difference between recursive function and pow() is 0
a =
2.3000
n =
100
a^n using recursive function is 1.488619e+36
difference between recursive function and pow() is 5.902958e+20
```

Q4

```
function r = my_mod_exp(a,n,p)
    if(n == 0)
        k = 1;
    else
        k = a*my_pow(a,n-1);
    end
    r = mod(k,p);
end

    a = input("Enter number: ")
```

```
a =
2
```

```
    n = input("Enter its exponent: ")
```

```
n =
3
```

```
    p = input("Enter its mod: ")
```

```
p =
5
```

```
    fprintf("a^n & m using recursive modular exponentiation function is
%d",my_mod_exp(a,n,p))
```

```
a^n & m using recursive modular exponentiation function is 3
```

Q5

```
function r = my_nth_fibonacci(n)
    if(n == 0)
        r = 0;
    elseif(n == 1)
            r = 1;
    else
        r = my_nth_fibonacci(n-1)+my_nth_fibonacci(n-2);
    end
end

n = input("Enter number: ")
```

```
n =
4
```

```
fprintf("%dth fibonacci number is %d",n,my_nth_fibonacci(n))
```

```
4th fibonacci number is 3
```

Q6

```
function r = my_nth_sequence(n)
    if(n == 0)
        r = 6;
    else
        r = 3*my_nth_sequence(n-1)+1;
    end
end

n = input("Enter number: ")
```

```
n =
5
```

```
fprintf("%dth sequence number is %d",n,my_nth_sequence(n))
```

```
5th sequence number is 1579
```

Q7

```
function r = my_inv_pow(a,n)
     if(n == 0)
        r = 1;
    else
        r = my_inv_pow(a,n+1) / a;
    end
end
    a = input("Enter number: ")
```

```
a =
2
```

```
    n = input("Enter its exponent (negative): ")
```

```
n =
-2
```

```
    fprintf("a^-n using recursive function is %d",my_inv_pow(a,n))
```

```
a^-n using recursive function is 2.500000e-01
```