



Sheet 5 (Recursion)

Class Work

- 1- Write a function that gets a number and returns its factorial using:
 - a) Iterative method.
 - b) Recursive method.
- 2- Write a function that gets two numbers and returns their multiplication with repeated addition using:
 - a) Iterative method.
 - b) Recursive method.

Home Work

- 1- Write a program that contains a **Recursive** function, **GCD**, that calculates the Greatest Common Divisor using the following method.

$$GCD(m,n) \begin{cases} m & \text{if } n=0 \\ GCD(n, m\%n) & \text{if } n>0 \end{cases}$$

- 2- Write a program that contains a Recursive function, Power (x, y), that calculates the value of a float number x raised to the power y (x^y), where y is a positive integer number.
 - Modify the Power function to calculate the value of x^{-y} .
 - Modify the Power function to calculate the value of (x^y) , where y has a positive or negative value .
- 3- Write a program that contains a **Recursive** function, **Fibo**, that calculates the Fibonacci series with the following function.

$$F(n) \begin{cases} 1 & \text{if } n = 0, 1 \\ F(n-1)+F(n-2) & \text{if } n > 1 \end{cases}$$

4- Trace the following programs:

```
#include<stdio.h>

int fun (int x)
{   If (x<1) return(1);
    Else
        Printf(“ %d %d \n”, x, fun(x-1));
}
void main()
{
int x,y;
x = 5;
y = fun(x);
printf(“\n x = %d f(x) = %d \n”, x, y);
}
```

```
#include<stdio.h>

int mystery (int n)
{
if (n<=0)
    return 1;
else
    return ( 3 + mystery (n-1));
}
void main()
{   int i;
    for (i=0; i<5; i++)
        printf (“\n i = %d \t value = %d\n”, i , mystery(i) )
}
```

```
#include<stdio.h>

int x=5;

void sum (int y)
{   int z;
    z = x + y;
    printf(“\n x= %d y= %d z=%d \n”, x++, y++, z++ );
}

void main()
{
int w=5;
sun(w);
printf(“\n x= %d w=%d \n”, x, w );
}
```