Sheet 8:

1- Complete the following:

- A program module in C is called a ………………………
- A function is invoked with a ………………….
- A variable that is known only within the function in which it is defined is called a …………..
- The …………… statement in a called function is used to pass the value of an expression back to the calling function.
- The keyword …………… is used in a function header to indicate that it does not return a value.
- The items passed in a function call are the ……………… but the corresponding ………… appear in the function prototype.
- Constants and expressions can be actual arguments corresponding to formal parameters that are …………… parameters.
- If an actual argument of –25.7 is passed to a type int formal parameter, what will happen?
- If an actual argument of 17 is passed to a type double formal parameter, what will happen?
- Formal parameters that are output parameters must have actual arguments that are …………..

2- What is the output of the following C programs:

```c
#include <stdio.h>    #include <stdio.h>
#include <math.h>    #include <math.h>
void main ( )                void Separt(double N , char *S, int *W , double *F)
{              {
    int x , z ;     double mag;
    x = 1 ;      if ( N < 0 )
    z = 10 ;        *S = '-' ;
print("Early %d%\n", x , z);             else if ( N == 0 )
OnTime(&z , x);       *S = ' ' ;
}                 else *S = '+' ;
void OnTime(int *w , int y)
{      mag = fabs( N ) ;
    *w = *w + 1;     *W=floor(mag) ;
    y = y + 2;              *F = mag - *W ;
printf("On Time %d%\n", *w , y);
}

void Separt(double v , char *sn , int &whlea , &frac ) ;
printf( "Sign : %c\n", sn ) ;
printf( "Whole Number Magnitude : %d\n",wh) ;
printf("Fraction Part : %.4f\n", frac ) ;
}
```

```c
int x , z ;
x = 1 ;
z = 10 ;
printf("Early %d%\n", x , z);
OnTime(&z , x);
void OnTime(int *w , int y)
{
    *w = *w + 1;
y = y + 2;
printf("On Time %d%\n", *w , y);
}
```

Dr. Magdy A. Ahmed
3- Write a function that allow either a real or an integer-type number to be raised to an integer-type power. In other word we wish to evaluate the formula \( y = x^n \) where \( y \) and \( x \) are either real or integer-type variables and \( n \) is an integer-type variable, and then, write a complete C program that will read in \( x \) and \( n \), then evaluate the formula using the above function.

4- Write a function Multiple that determines for a pair of integers whether the second integer is a multiple of the first. The function should take two integer arguments and return 1 (true) if the second is a multiple of the first, and 0 (false) otherwise. Use this function in a program that inputs a series of pairs of integers.

5- Write a function that counts the number of digits in a number.

6- Write a function called Letter_Grade that has a type int parameter called points and returns the appropriate letter grade using a straight scale (90-100 is an A, 80-90 is an B, and so on).

7- Write a function named Prime that checks a positive integer to see whether it is prime number. Your function should return a 1 for true if its argument is prime, and a 0 for false if it is not.

8- Write a complete C program that will calculate the roots of the quadratic equation \( ax^2 + bx + c = 0 \) Modularize your program, one for real roots, and one for complex roots.

9- Write a function that takes \( N \) as an input and finds the sum of numbers from 1 to \( N \), odd numbers from 1 to \( N \) and even numbers from 1 to \( N \).

10- Write a function that swaps two integer numbers. Swap(int *x , int *y)

11- Write a modular C program to display the following figure. (bonus)