

# PROGRAMACIÓN

**Unidad 4:** Funciones de biblioteca. Biblioteca estándar. El prototipo de una función. Funciones parametrizadas.

# Repasemos lo visto



# Funciones de Biblioteca

- ❑ C proporciona funciones predefinidas que no forman parte del lenguaje (denominadas funciones de biblioteca). Se invocan por su nombre y los parámetros opcionales que incluye. Después de que la función sea llamada, el código asociado con la función se ejecuta y, a continuación, se retorna el valor obtenido.
- ❑ En C, a diferencia de las funciones definidas por el usuario que requieren una declaración o prototipo en el programa, que indica al compilador el nombre por el cual ésta será invocada, el tipo y el número y tipo de sus argumentos, las funciones de biblioteca requieren que se incluya el archivo donde está su declaración.

# Ejemplo

**Biblioteca donde se encuentra  
la declaración de las funciones  
de entrada y salida**

```
1  #include<stdio.h>
2
3  int main(void)
4  {
5      int edad;
6
7      printf("Ingrese su edad: "); Función de Salida
8
9      scanf("%d", &edad); Función de Entrada
10
11     printf("La edad ingresada es: %d", edad);
12     return 0;
13 }
```

# Ejemplo

En el siguiente ejemplo, de la función seno, de la biblioteca math.h

```
1  #include<stdio.h>
2  #include<math.h>
3
4  int main(void)
5  {
6      double result, x = 0.5;
7      int i;
8
9      for(i=1;i<=20;i++)
10     {
11         result=sin(x);
12         x = x+0.5;
13         printf("El seno() de %lf es %lf\n", x, result);
14     }
15     return 0;
16 }
```

**Biblioteca donde se encuentra la declaración de las funciones matemáticas**

**Función seno**

# Funciones de Biblioteca

Algunas funciones de biblioteca que podemos mencionar son:

**<stdio.h>**

**printf()**  
**scanf()**  
**getchar()**            `car = getchar();`  
**putchar()**            `putchar(car);`

**<math.h>**

**sqrt(x)**: raiz cuadrada de x  
**pow(x,y)**: x elevado a la potencia y  
**sin(x)**: seno de x expresado en radianes  
**cos(x)**: coseno de x expresado en radianes

**<ctype.h>**

**isalpha(c)**: retorna verdadero si c es una letra mayúscula o minúscula. Retorna falso si c no es letra.  
**islower(c)**: retorna verdadero si c es una letra minúscula. Retorna falso si c no es una letra minúscula.  
**isupper(c)**: retorna verdadero si c es una letra mayúscula. Retorna falso si c no es una letra mayúscula.

# Funciones de Biblioteca

En la tabla,  $x$  e  $y$  son de tipo double, las funciones regresan double. Los ángulos para las funciones trigonométricas están expresados en radianes.

Función	Descripción	Ejemplo de uso
sqrt	raíz cuadrada de $x$	sqrt (900.0) es 30.0
exp( $x$ )	función exponencial $e^x$	exp(1.0) es 2.718282 exp(2.0) es 7.389056
log( $x$ )	logaritmo natural de $x$ (base $e$ )	log(2.718282) es 1.0 log(7.389056) es 2.0
log10( $x$ )	logaritmo de $x$ (base 10)	log10(1.0) es 0.0 log10(100) es 2.0
ceil( $x$ )	redondea a $x$ al entero más pequeño que no sea menor que $x$	ceil(9.2) es 9.0 ceil(-9.8) es -10.0
floor( $x$ )	redondea a $x$ al entero más grande no mayor que $x$	floor(9.2) es 9.0 floor(-9.8) es -9.0
pow( $x$ , $y$ )	$x$ elevado a la potencia $y$	pow(2.7) es 128.0
sin( $x$ )	seno de $x$ expresado en radianes	sin(0.0) es 0.0
cos( $x$ )	coseno de $x$ expresado en radianes	cos(0.0) es 1.0
tan( $x$ )	tangente de $x$ expresado en radianes	tan(0.0) es 0.0

# Funciones de Biblioteca

Funciones de uso  
común de la  
biblioteca de manejo  
de caracteres:  
<ctype.h>

Función	Descripción
int isdigit (int c)	SI (c es un dígito) ENTONCES regresa un valor verdadero SINO regresa un valor falso
int isalpha (int c)	SI (c es una letra) ENTONCES regresa un valor verdadero SINO regresa un valor falso
int isalnum (int c)	SI (c es un dígito o una letra) ENTONCES regresa un valor verdadero SINO regresa un valor falso
int islower (int c)	SI (c es una letra minúscula) ENTONCES regresa un valor verdadero SINO regresa un valor falso
int isupper (int c)	SI (c es una letra mayúscula) ENTONCES regresa un valor verdadero SINO regresa un valor falso
int tolower (int c)	SI (c es una letra mayúscula) ENTONCES regresa c como una letra minúscula SINO regresa c sin cambios
int toupper (int c)	SI (c es una letra minúscula) ENTONCES regresa c como una letra mayúscula SINO regresa c sin cambios



# Funciones de Biblioteca

¿Podemos nosotros crear nuestra propia biblioteca con funciones personalizadas?

La respuesta a esa pregunta es **SI**, podemos crear una biblioteca con nuestras funciones (Funciones definidas por el usuario). Explicaremos mediante los pasos a seguir mediante el siguiente ejemplo:

**Realice una función que dado dos números enteros devuelva el resultado de su suma.**

# Funciones de Biblioteca

## Opción N° 1

- 1 Escribir el prototipo de la función y la definición de la misma.

```
1  int suma(int x, int y); // prototipo de la función
2
3
4  int suma(x,y)
5  {
6      return(x+y); //retorno de la función
7  }
```

- 2 Guarde el archivo con el nombre: misFunciones.h, en el lugar donde se instalo el compilador. Por ejemplo: **C:\MinGW\include**

- 3 Ahora usemos nuestra librería...

```
1  #include <stdio.h>
2  #include <misFunciones.h>
3
4  int main(void)
5  {
6      int num1 = 5, num2 = 10, resultado;
7
8      resultado = suma(num1, num2);
9      printf("El resultado es: %d", resultado);
10
11     return 0;
12 }
```

# Funciones de Biblioteca

## Opción N° 2

- 1 Escribir el prototipo de la función y la definición de la misma.

```
1  int suma(int x, int y); // prototipo de la función
2
3
4  int suma(x,y)
5  {
6      return(x+y); //retorno de la función
7  }
```

- 2 Guarde el archivo con el nombre: misFunciones.h, en el mismo lugar donde se encuentra su código .c principal

- 3 En este caso para hacer uso de esa librería debe incluirla usando comillas.

```
1  #include <stdio.h>
2  #include "misFunciones.h"
3
4  int main(void)
5  {
6      int num1 = 5, num2 = 10, resultado;
7
8      resultado = suma(num1, num2);
9      printf("El resultado es: %d", resultado);
10
11     return 0;
12 }
```

```
#include <stdio.h>
```

```
int main()
```

```
{
```

```
    printf("Hasta la próxima clase!!\n");
```

```
    return 0;
```

```
}
```