



## Carátula para entrega de prácticas

Facultad de Ingeniería

Laboratorio de docencia

# Laboratorio de computación salas A y B

*Profesor:*

Alejandro Esteban Pimentel

*Asignatura:*

Fundamentos de programación

*Grupo:*

Bloque 135

*No de Práctica(s):*

12

*Integrante(s):*

Gutierrez Acosta Claudia

*No. de Equipo empleado:*

42

*No. de Lista o Brigada:*

2881

*Semestre:*

Primer Semestre

*Fecha de entrega:*

04/11/2019

*Observaciones:*

**CALIFICACIÓN:**

\_\_\_\_\_

# FUNCIONES

## Introducción:

En esta ocasión utilizaremos diferentes funciones para hacer más sencillos los trabajos de programación en lenguaje C y conocer más sobre este pues al implementar estas características los programas serán mejor ejecutados.

## Objetivo:

Elaborar programas en C donde la solución del problema se divida en funciones. Distinguir lo que es el prototipo o firma de una función y la implementación de ella, así como manipular parámetros tanto en la función principal como en otras.

## Funciones

```
valor Retorno nombre (parámetros) {  
    // bloque de código de la función  
}
```

## Actividades

Las actividades deben tener los prototipos de sus funciones, y sus funciones implementadas después del *main*.

- Crear un programa que tenga una función que regrese el factorial de un número de entrada.
- Crear un programa que tenga una función que regrese el resultado de la serie:

$$\sum_{x=1}^n \frac{x!}{x}$$

Para un número  $n$  de entrada. Utilizar la función de factorial de la primera actividad.

## Actividad 1

```
1  #include<stdio.h>
2  double factorial (long int);
3  int main(){
4      long int x, resultado, y;
5      printf("ingrese un valor");
6      scanf("%li", &x);
7      y=x;
8      resultado=factorial(x);
9      printf("el factorial de %li es %li\n", y, resultado);
10     return 0;
11 }
12 double factorial (long int x){
13     long int resultado=1;
14     while (x>1){
15         resultado=resultado*x;
16         x--;
17     }
18     return resultado;
19 }
```

Compilation results...

- Errors: 0
- Warnings: 0
- Output Filename: C:\Users\Fabiola\Pictures\7\practical21.exe
- Output Size: 128.62890625 KiB
- Compilation Time: 3.36s

Para un número n de entrada. Utilizar la función de factorial de la primera actividad

```
C:\Users\Fabiola\Pictures\7\practical21.exe
ingrese un valor 8
el factorial de 8 es 40320
-----
Process exited after 3.961 seconds with return value 0
Presione una tecla para continuar . . .
```

## Actividad 2:

```
1  #include<stdio.h>
2  #include<math.h>
3  int a;
4  unsigned long long resultado;
5  unsigned long long int factorial (int a){
6      resultado=a;
7      for(int b=1; b<a; b++){
8          resultado=resultado*b;
9      }
10     return resultado;
11 }
12 int main (){
13     int n;
14     printf("ingrese cuántos términos calcular de la sumatoria: X!/X\n");
15     printf("n=");
16     scanf("%i", &n);
17 }
18 int suma (){
19     unsigned long long int suma;
20     int n;
21     suma=factorial(1)/1;
22     for (a=2;a<=n;a++){
23         suma=suma+ factroial(a)/a;
24     }
25     printf("el resultado de la sumatoria es: %llu", suma);
26     return 0;
27 }
28
29
```

## CONCLUSION

Con la investigación previa que hice sobre este tema encontré que las funciones pueden tomar parámetros que modifiquen su funcionamiento. Las funciones son utilizadas para descomponer grandes problemas en tareas simples y para implementar operaciones que son comúnmente utilizadas durante un programa y de esta manera reducir la cantidad de código. Cuando una función es invocada se le pasa el control a la misma, una vez que esta finalizó con su tarea el control es devuelto al punto desde el cual la función fue llamada.