



Carátula para entrega de prácticas

Facultad de Ingeniería

Laboratorio de docencia

Laboratorios de computación salas A y B

<u>Profesor:</u>	Ing. Karina García Morales
<u>Asignatura:</u>	Fundamentos de Programación
<u>Grupo:</u>	1121
<u>No de Práctica(s):</u>	6
<u>Integrante(s):</u>	Jacinto Rodriguez Moisés Rodrigo
<u>No. de Equipo de cómputo empleado</u>	6
<u>Semestre:</u>	2019-1
<u>Fecha de entrega:</u>	3/10/18
<u>Observaciones:</u>	

CALIFICACIÓN: _____

Practica 6 Entorno de C (editores, compilación y ejecución)

Objetivo

Conocer y usar los ambientes y herramientas para el desarrollo y ejecución de programas en Lenguaje C, como editores y compiladores en diversos sistemas operativos.

Desarrollo

NOTA: las instrucciones de compilar un programa

¿Por qué se usa mucho el lenguaje C?

Porque es un lenguaje de alto nivel y es uno de los más cercanos al lenguaje comúnmente usados por las personas.

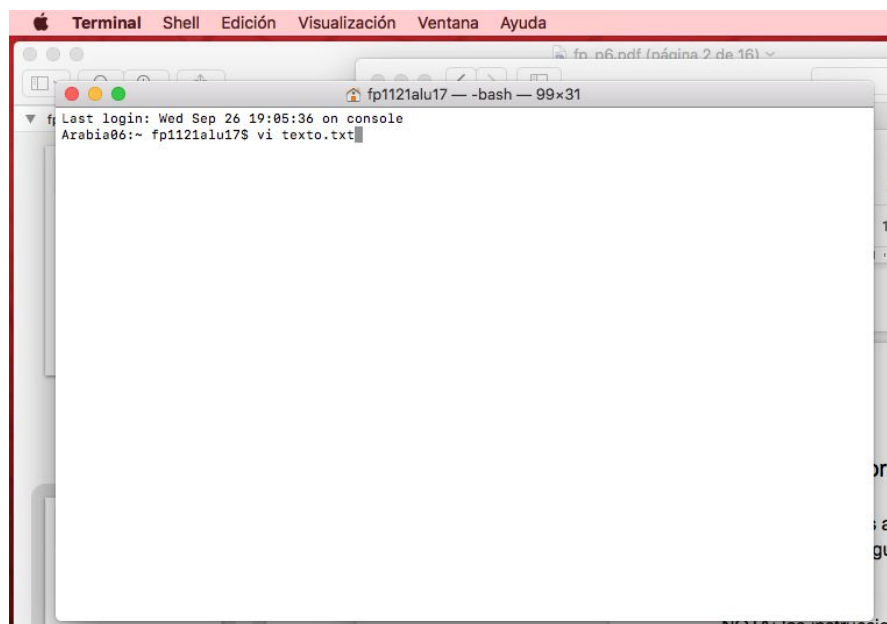
Código fuente → compilador → ejecutable: objeto o .exe

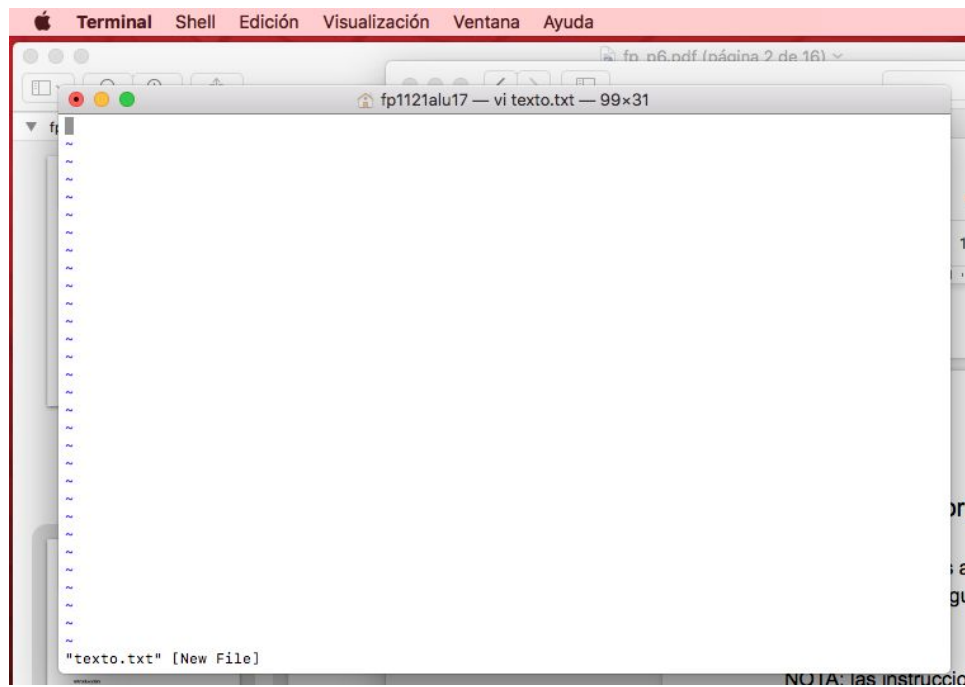
Editor de texto → Texto plano (bloc de notas o notepad) → Vision interface, nano, gedit

Word genera más información; se necesita un texto plano porque pasa por un compilador y es más ligero

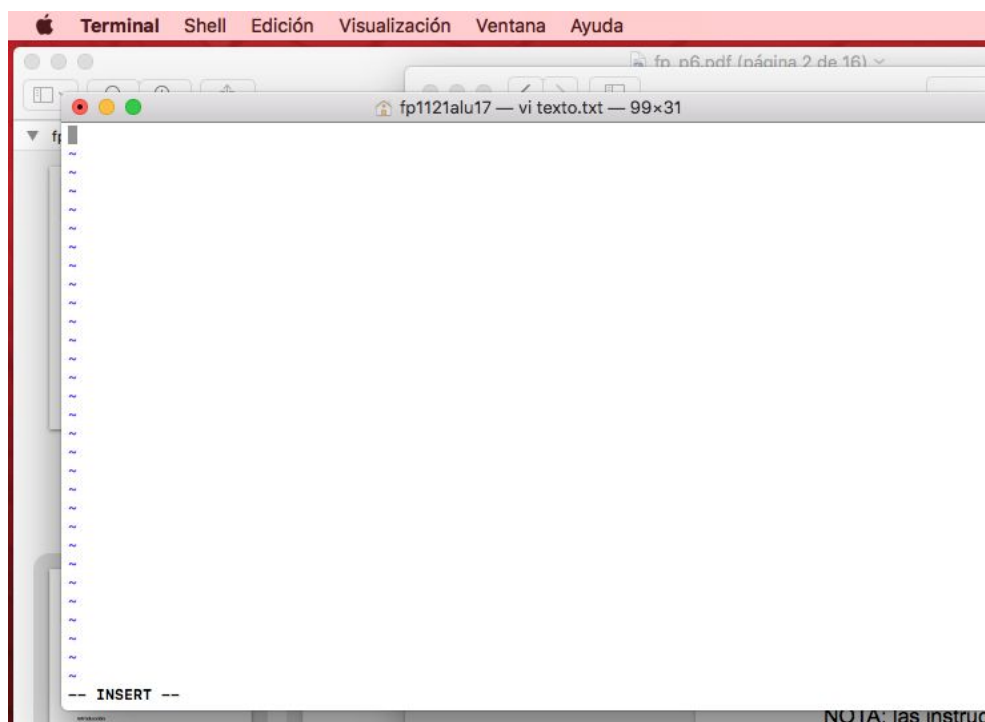
Procesador de texto → Formato al texto que se ingresa, imágenes

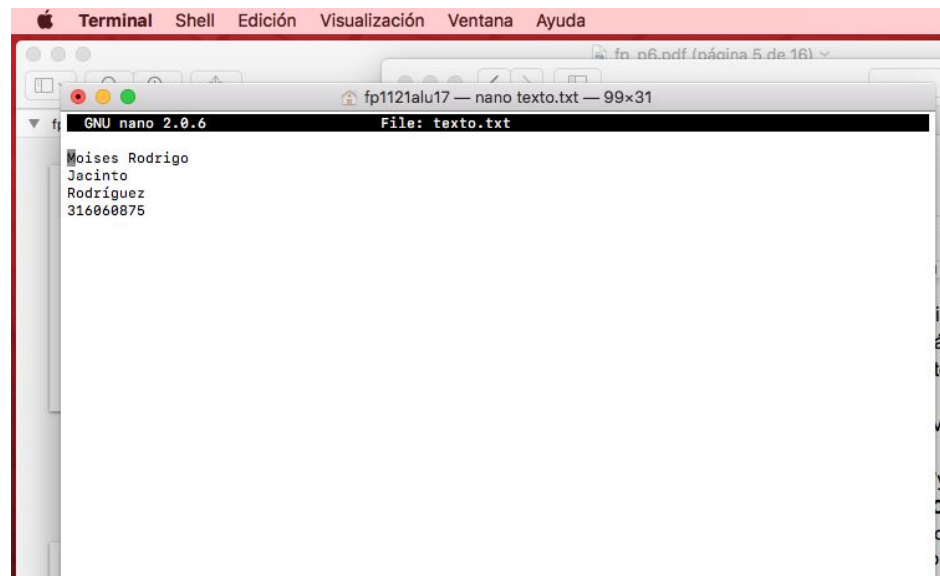
Crear o abrir archivos





Esc → i → Editar





↑ o **k** mueve el puntero hacia arriba.

↓ o **j** mueve el puntero hacia abajo.

← o **h** mueve el puntero hacia la izquierda.

→ o **l** mueve el puntero hacia la derecha.

1G lleva el puntero al comienzo de la primera línea.

G lleva el puntero al inicio de la última línea.

x borra el carácter marcado por el cursor.

dd elimina o quita la línea donde está el puntero.

ndd donde **n** es la cantidad de líneas que se eliminarán o quitarán después del puntero.

D elimina o quita desde la posición de puntero hasta el final de la línea.

dw elimina o quita desde la posición del puntero hasta el final de una palabra.

yy copia la línea donde está el puntero.

p pega un contenido copiado o borrado.

u deshace el último cambio.

Esc: q → Salir

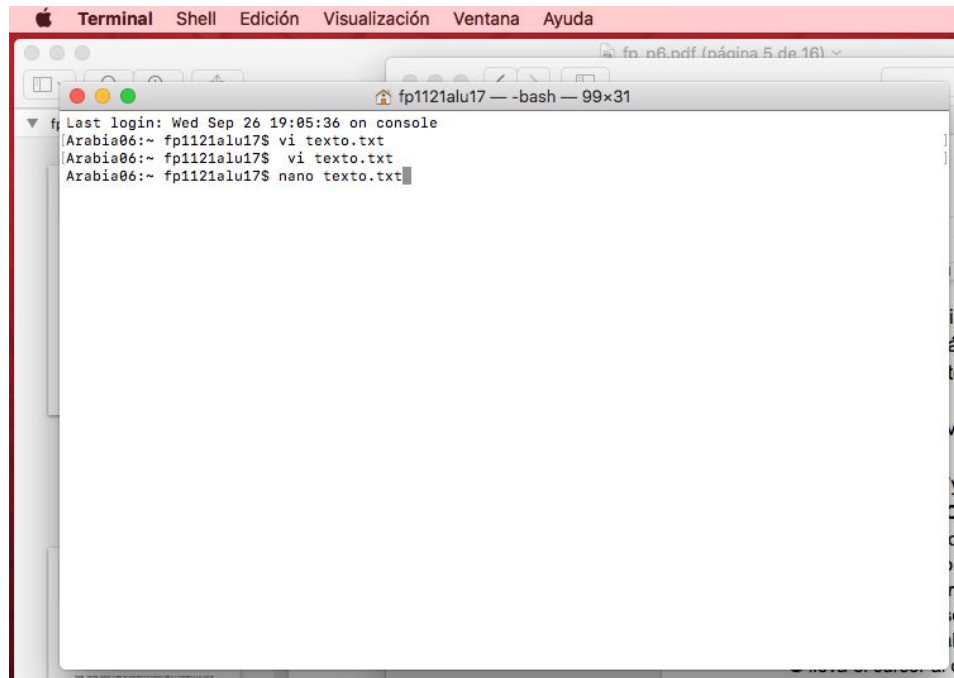
Esc: wq → Guardar y salir

Esc: wq! → Fuerzan salida y guardado

Esc A → Mueve el cursor y empieza a insertar información

nano texto.txt *Captura*

Control X + N → Salir *Captura*



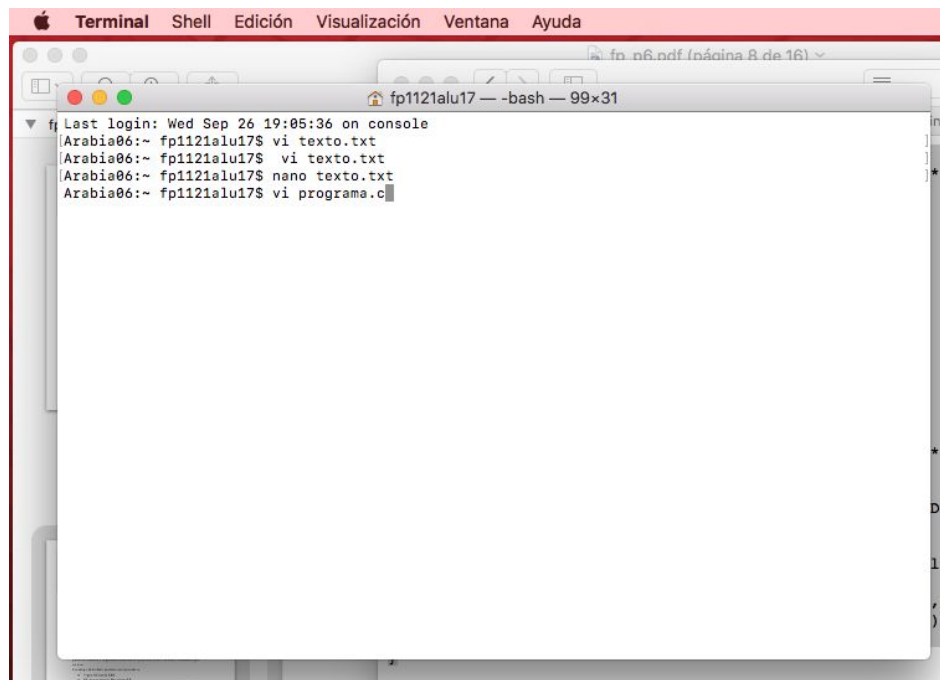
Si se utilizan gráficos se terminan los recursos del equipo
¿Por qué el procesador de texto no es plano y no lo usamos?
genera datos y códigos que no vemos

¿Qué hace un compilador en esos programas?
Valida que la sintaxis sea la correcta al ejecutar algún proceso y suele mandar en la salida cuál fue el error

gcc calculadora.c genera a.out
Es más problemático porque es compilar → ejecutar las veces que sea necesario

El compilador gcc programa.c -o programa.out es más eficaz

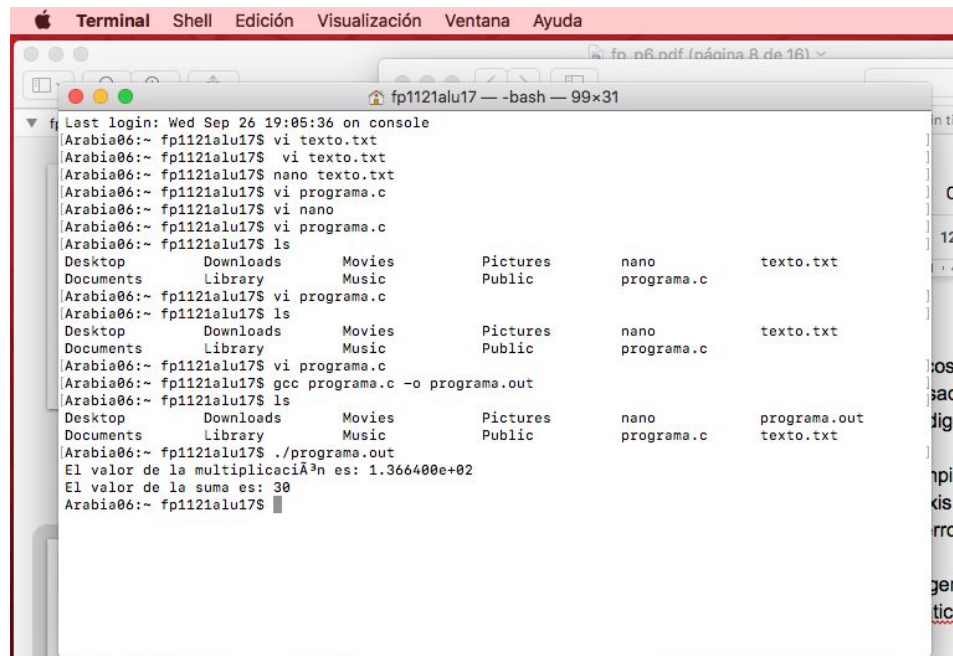
1.- creación del archivo
“ vi programa.c



2.- compilación del archivo se usa el adicional “ls” para verificar que si se encuentra gcc programa.c -o programa.out



3.- ./programa.out + Enter



```
Terminal Shell Edición Visualización Ventana Ayuda
fp1121alu17 — -bash — 99x31
Last login: Wed Sep 26 19:05:36 on console
Arabia06:~ fp1121alu17$ vi texto.txt
Arabia06:~ fp1121alu17$ vi texto.txt
Arabia06:~ fp1121alu17$ nano texto.txt
Arabia06:~ fp1121alu17$ vi programa.c
Arabia06:~ fp1121alu17$ vi nano
Arabia06:~ fp1121alu17$ vi programa.c
Arabia06:~ fp1121alu17$ ls
Desktop    Downloads  Movies     Pictures   nano       texto.txt
Documents  Library    Music      Public     programa.c
Arabia06:~ fp1121alu17$ vi programa.c
Arabia06:~ fp1121alu17$ ls
Desktop    Downloads  Movies     Pictures   nano       texto.txt
Documents  Library    Music      Public     programa.c
Arabia06:~ fp1121alu17$ vi programa.c
Arabia06:~ fp1121alu17$ gcc programa.c -o programa.out
Arabia06:~ fp1121alu17$ ls
Desktop    Downloads  Movies     Pictures   nano       programa.out
Documents  Library    Music      Public     programa.c texto.txt
Arabia06:~ fp1121alu17$ ./programa.out
El valor de la multiplicación es: 1.366400e+02
El valor de la suma es: 30
Arabia06:~ fp1121alu17$
```

TAREA

1. Calcular la suma de los primeros 20 números pares

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

```
int main(){
```

```
    int i,suma = 0;
```

```
    for(i=1;i<=40;i++){
```

```
        if(i%2==0){
```

```
            suma = suma + i;
```

```
        }
```

```
    }
```

```
    printf("\n La suma de los numeros pares es: %i",suma);
```

```
}
```

```
fp1121alu17 — vi rodri.c — 80x24
#include<stdio.h>

int main(){
    int i,suma = 0;

    for(i=1;i<=40;i++){
        if(i%2==0){

            suma = suma + i;
        }
    }

    printf("\n La suma de los numeros pares es: %i",suma);
}

"rodri.c" 15L, 177C
```

```
fp1121alu17 — -bash — 80x24
[Argelia07:~ fp1121alu17$ vi rodri.c
[Argelia07:~ fp1121alu17$ gcc rodri.c -o rodri.out
[Argelia07:~ fp1121alu17$ ./rodri.out

La suma de los numeros pares es: 420Argelia07:~ fp1121alu17$
```


2.- Realizar un programa que lea x cantidad de números, hasta que su suma sea mayor a 200. Imprimir al final, la cantidad de números leídos, sus suma y promedio.

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

```
int main(){
```

```
    int i, suma = 0;
```

```
    float promedio;
```

```
    promedio = suma / 20;
```

```
    for(i=1;i<=20;i++){
```

```
        suma = suma + i;
```

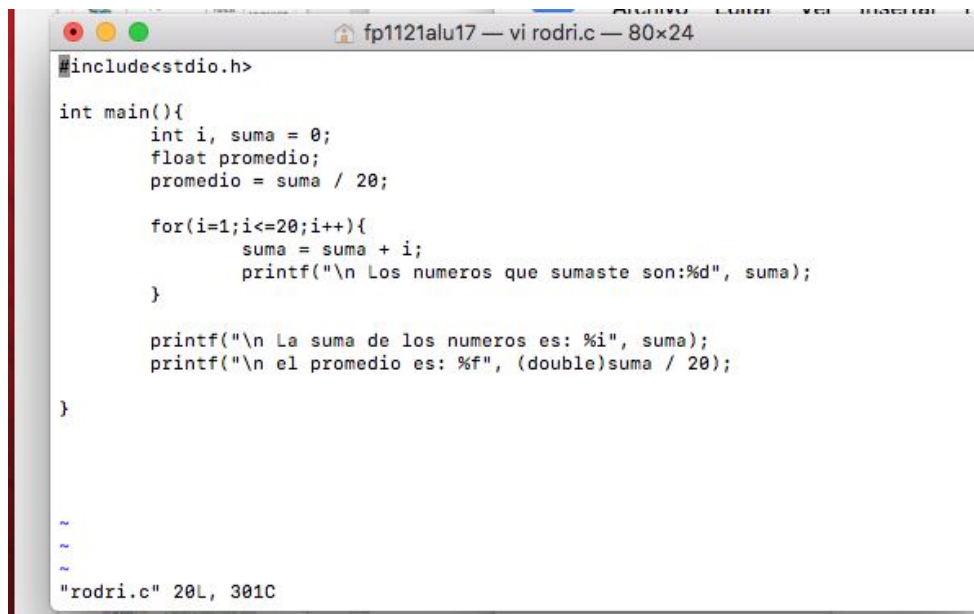
```
        printf("\n Los numeros que sumaste son:%d", suma);
```

```
    }
```

```
    printf("\n La suma de los numeros es: %i", suma);
```

```
    printf("\n el promedio es: %f", (double)suma / 20);
```

```
}
```

A screenshot of a terminal window with a title bar that reads "fp1121alu17 — vi rodri.c — 80x24". The terminal displays the C code from the previous blocks. At the bottom of the terminal, it shows the file name and line/column information: "rodri.c" 20L, 301C.

```
#include<stdio.h>

int main(){
    int i, suma = 0;
    float promedio;
    promedio = suma / 20;

    for(i=1;i<=20;i++){
        suma = suma + i;
        printf("\n Los numeros que sumaste son:%d", suma);
    }

    printf("\n La suma de los numeros es: %i", suma);
    printf("\n el promedio es: %f", (double)suma / 20);

}
```

"rodri.c" 20L, 301C

```
fp1121alu17 — -bash — 80x24
Argelia07:~ fp1121alu17$ ./rodri.out
Los numeros que sumaste son:1
Los numeros que sumaste son:3
Los numeros que sumaste son:6
Los numeros que sumaste son:10
Los numeros que sumaste son:15
Los numeros que sumaste son:21
Los numeros que sumaste son:28
Los numeros que sumaste son:36
Los numeros que sumaste son:45
Los numeros que sumaste son:55
Los numeros que sumaste son:66
Los numeros que sumaste son:78
Los numeros que sumaste son:91
Los numeros que sumaste son:105
Los numeros que sumaste son:120
Los numeros que sumaste son:136
Los numeros que sumaste son:153
Los numeros que sumaste son:171
Los numeros que sumaste son:190
Los numeros que sumaste son:210
La suma de los numeros es: 210
el promedio es: 10.500000Argelia07:~ fp1121alu17$
```

4.-

```
#include<stdio.h>
#include<stdlib.h>
#include<time.h>
int main()
{
    float importeCompra,descuento;
    int num, pctDescuento;
    srand(time(NULL));
    num=rand() % 8; // rango de valores
    printf ("\n-----");
    switch(num)
    {
        case 0:
            printf ("\nBolita de color negro");
            descuento=10;
            break;
        case 1: printf ("\nBolita de color verde"); descuento=25;
            break;
        case 2: printf ("\nBolita de color amarillo"); descuento=50;
            break;
        case 3: printf ("\nBolita de color azul"); descuento=75;
            break;
```

```

        case 4: printf ("\nBolita de color rojo"); descuento=100;
        break;
        default:
                descuento=0;
                printf ("\n Sin descuento");
    }
    printf ("\n\nSu descuento es de %f porciento", descuento);
    printf ("\n-----");
    printf ("\nImporte total de la compra (sin descuento): $ ");
    scanf ("%4.2f", &importeCompra);
    descuento = importeCompra * ( 1 - descuento / 100);
    printf ("\nSu importe total de compra fue: %4.2f", importeCompra);
    printf ("\nEl importe total con el descuento será de: $ %.2f \n\n",
descuento);
return 1;
}

```

Conclusión

El lenguaje C es uno de los más fundamentales en el terreno de la programación, ya que conforma uno de los pilares básicos de la computación y automatización; cabe destacar que hay distintos programas para dar solución a una serie de problemas manifestados, pero este lenguaje es utilizado cuando se requiere hacer una tarea con funciones básicas fundamentadas de manera satisfactoria.

Las actividades llevadas a cabo dentro de esta práctica dan una introducción para conocer paso a paso el funcionamiento de los elementos más esenciales de este lenguaje, para entenderlos con mayor claridad con ayuda de trabajos pasados para comprender el funcionamiento correcto de esta útil herramienta en la vida ingenieril.