



## Carátula para entrega de prácticas

Facultad de Ingeniería

Laboratorio de docencia

# Laboratorios de computación salas A y B

*Profesor:* M. C. ALEJANDRO ESTEBAN PIMENTEL ALARCON

*Asignatura:* FUNDAMENTOS DE PROGRAMACION

*Grupo:* 3

*No de Práctica(s):* 10

*Integrante(s):* MARTINEZ TRUJILLO CINTHYA ANDREA

*No. de Equipo de  
cómputo empleado:*

*No. de Lista o Brigada:* 2720

*Semestre:* 2020-1

*Fecha de entrega:* 21 / OCTUBRE / 2019

*Observaciones:* Tarde entrega.  
No se cumplen con los objetivos de las primeras  
dos actividades, no está la descripción que se pide  
en la primera, ni corrección en la segunda.

CALIFICACIÓN: 4

# DEPURACION DE PROGRAMAS

## INTRODUCCION

En esta practica usaremos linux ya que el comando gdb en mac no es reconocido, observaremos como es el proceso de identificacion y correccion de errores en programacion.

## OBJETIVO

Aprender las tecnicas basicas de depuracion de programas en C para revisar de manera precisa el flujo de ejecucion de un programa y el valor de las variables; en su caso, corregir posibles errores.

## EJEMPLO

```
#include <stdio.h>

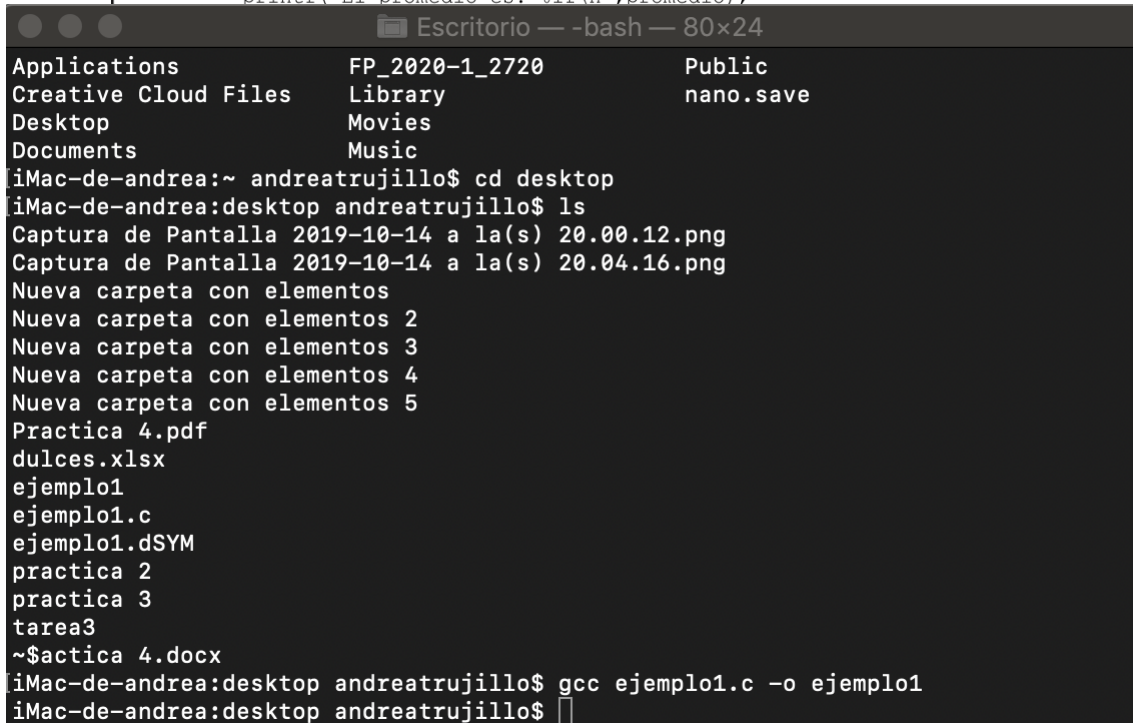
int main(int argc, char * argv[]) {

    // Asignamos variables
    int numero = 10;
    int lista[numero];
    char caracter = 'B';
    float numeroReal = 89.8;
    long int suma = 0;
    double promedio;

    // Mostramos texto y valores
    printf("Primero texto solo\n");
    printf("Luego podemos poner un entero: %i\n", numero);
    printf("También podemos poner un caracter: %c\n", caracter);
    printf("Un numero real: %.2f\n", numeroReal);

    // Podemos llenar la lista con valores
    for(int i = numero ; i >= numero ; i++){
        lista[i] = i;
    }

    // Y ahora podemos hacer calculos con la lista
    for(int i = numero ; i >= numero ; i++){
        suma += lista[i];
    }
    promedio = suma / numero;
    printf("La suma es: %li\n", suma);
    printf("El promedio es: %lf\n", promedio);
}
```



The screenshot shows a terminal window titled "Escritorio — -bash — 80x24". On the left, a file explorer sidebar is visible with categories: Applications, Creative Cloud Files, Desktop, Documents, Library, Movies, Music, and Public. The main terminal area shows the following commands and output:

```
iMac-de-andrea:~ andreatrujillo$ cd desktop
iMac-de-andrea:desktop andreatrujillo$ ls
Captura de Pantalla 2019-10-14 a la(s) 20.00.12.png
Captura de Pantalla 2019-10-14 a la(s) 20.04.16.png
Nueva carpeta con elementos
Nueva carpeta con elementos 2
Nueva carpeta con elementos 3
Nueva carpeta con elementos 4
Nueva carpeta con elementos 5
Practica 4.pdf
dulces.xlsx
ejemplo1
ejemplo1.c
ejemplo1.dSYM
practica 2
practica 3
tarea3
~$actica 4.docx
iMac-de-andrea:desktop andreatrujillo$ gcc ejemplo1.c -o ejemplo1
iMac-de-andrea:desktop andreatrujillo$
```

```
Escritorio — -bash — 80x24
Captura de Pantalla 2019-10-14 a la(s) 20.00.12.png
Captura de Pantalla 2019-10-14 a la(s) 20.04.16.png
Nueva carpeta con elementos
Nueva carpeta con elementos 2
Nueva carpeta con elementos 3
Nueva carpeta con elementos 4
Nueva carpeta con elementos 5
Practica 4.pdf
dulces.xlsx
ejemplo1
ejemplo1.c
ejemplo1.dSYM
practica 2
practica 3
tarea3
~$actica 4.docx
iMac-de-andrea:desktop andreatrujillo$ gcc ejemplo1.c -o ejemplo1
iMac-de-andrea:desktop andreatrujillo$ ./ejemplo1
Primer texto solo
Luego podemos poner un entero: 10
También podemos poner un caracter: B
Un numero real: 89.80
Segmentation fault: 11
iMac-de-andrea:desktop andreatrujillo$
```

## quit (q)

```
fp03alu27 — fp03alu27@samba:~ — ssh fp03alu27@192.168.2.200 — 80x24
Un numero real: 89.80

Program received signal SIGSEGV, Segmentation fault.
0x0000000040060c in main (argc=19, argv=0x110000010) at ejemplo1.c:21
21             lista[i] = i;
Missing separate debuginfos, use: debuginfo-install glibc-2.15-37.fc17.x86_64
(gdb) list
16             printf("También podemos poner un caracter: %c\n", caracter);
17             printf("Un numero real: %.2f\n", numeroReal);
18
19             // Podemos llenar la lista con valores
20             for(int i = numero ; i >= numero ; i++){
21                 lista[i] = i;
22             }
23
24             // Y ahora podemos hacer calculos con la lista
25             for(int i = numero ; i >= numero ; i++){
(gdb) quit
A debugging session is active.

        Inferior 1 [process 21600] will be killed.

Quit anyway? (y or n) y
[fp03alu27@samba ~]$
```

## CTRL+X+A

```
fp03alu27 — fp03alu27@samba:~ — ssh fp03alu27@192.168.2.200 — 80x24
[ No Source Available ]

exec No process In:                               Line: ??  PC: ??
(gdb)
```

## Start

```
fp03alu27 — fp03alu27@samba:~ — ssh fp03alu27@192.168.2.200 — 80x24

ejemplo1.c
1 #include <stdio.h>
2
3 int main(int argc, char * argv[]) {
4
5     // Asignamos variables
6     int numero = 10;
7     int lista[numero];
8     char caracter = 'B';
9     float numeroReal = 89.8;
10    long int suma = 0;
11    double promedio;
12
13    // Mostramos texto y valores

child process 22121 In: main Line: 6 PC: 0x400542
(gdb) start
Temporary breakpoint 1 at 0x400542: file ejemplo1.c, line 6.
Starting program: /users/fp03/fp03alu27/ejemplo1

Temporary breakpoint 1, main (argc=1, argv=0x7ffffffe398) at ejemplo1.c:6
Missing separate debuginfos, use: debuginfo-install glibc-2.15-37.fc17.x86_64
(gdb)
```

## Actividad 1

Utilizar GDB para encontrar la utilidad del programa y describir su funcionalidad.

```
1 #include <stdio.h>
2
3 void main()
4 {
5     int N, CONT, AS;
6     AS=0;
7     CONT=1;
8     printf("Ingresa un número: ");
9     scanf("%i",&N);
10    while (CONT<=N)
11    {
12        AS=(AS+CONT);
13        CONT=(CONT+2);
14    }
15    printf("\nEl resultado es: %i\n", AS);
16 }
17
```

```
Julia@Julia-VirtualBox:~/Descargas$ gcc -g actividad1.c -o actividad1
Julia@Julia-VirtualBox:~/Descargas$ gdb ./actividad1
GNU gdb (Ubuntu 8.1-2ubuntu1) 8.1.0.20180409-git
Copyright (C) 2018 Free Software Foundation, Inc.
License GPLv3+: GNU GPL version 3 or later <http://gnu.org/licenses/gpl.html>
This is free software: you are free to change and redistribute it.
There is NO WARRANTY, to the extent permitted by law. Type "show copying"
and "show warranty" for details.
This GDB was configured as "x86_64-linux-gnu".
Type "show configuration" for configuration details.
Para las instrucciones de informe de errores, vea:
<http://www.gnu.org/software/gdb/bugs/>
Find the GDB manual and other documentation resources online at:
<http://www.gnu.org/software/gdb/documentation/>.
For help, type "help".
Type "apropos word" to search for commands related to "word"...
Legenda símbolos desde ./actividad1... hecho.
Julia@Julia-VirtualBox:~/Descargas$ gdb ./actividad1
GNU gdb (Ubuntu 8.1-2ubuntu1) 8.1.0.20180409-git
Copyright (C) 2018 Free Software Foundation, Inc.
License GPLv3+: GNU GPL version 3 or later <http://gnu.org/licenses/gpl.html>
This is free software: you are free to change and redistribute it.
There is NO WARRANTY, to the extent permitted by law. Type "show copying"
and "show warranty" for details.
This GDB was configured as "x86_64-linux-gnu".
Type "show configuration" for configuration details.
Para las instrucciones de informe de errores, vea:
<http://www.gnu.org/software/gdb/bugs/>
Find the GDB manual and other documentation resources online at:
<http://www.gnu.org/software/gdb/documentation/>.
For help, type "help".
Type "apropos word" to search for commands related to "word"...
Legenda símbolos desde ./actividad1... hecho.
(gdb) break 18
Breakpoint 1 at 0x708: file actividad1.c, line 18.
(gdb) run
Starting program: /home/julia/Descargas/actividad1
Ingresa un número: 6

Breakpoint 1, main () at actividad1.c:18
18     while (CONT<=N)
Julia@Julia-VirtualBox:~/Descargas$
```

```

Peppermint Terminal
actividad1.c
1 {
2     int N, CONT, AS;
3     AS=0;
4     CONT=1;
5     printf("Ingresa un número: ");
6     scanf("%i",&N);
7     while(CONT<=N)
8     {
9         AS=(AS+CONT);
10        CONT=(CONT+1);
11    }
12    printf("\nEl resultado es: %i\n", AS);
13 }
14
Native process 2078 In: main
(gdb) next
(gdb) print
La historia está vacía.
(gdb) print AS
$1 = 0
(gdb) print CONT
$2 = 1
(gdb) _

```

```

julio@julio-VirtualBox ~/Descargas $ ./actividad1
Ingresa un número: 1
El resultado es: 1
julio@julio-VirtualBox ~/Descargas $ ./actividad1
Ingresa un número: 2
El resultado es: 1
julio@julio-VirtualBox ~/Descargas $ ./actividad1
Ingresa un número: 3
El resultado es: 4
julio@julio-VirtualBox ~/Descargas $ ./actividad1
Ingresa un número: 4
El resultado es: 4
julio@julio-VirtualBox ~/Descargas $ ./actividad1
Ingresa un número: 5
El resultado es: 9
julio@julio-VirtualBox ~/Descargas $ ./actividad1
Ingresa un número: 6
El resultado es: 9
julio@julio-VirtualBox ~/Descargas $ ./actividad1
Ingresa un número: 7
El resultado es: 16
julio@julio-VirtualBox ~/Descargas $ ./actividad1
Ingresa un número: 8
El resultado es: 16

```

## Actividad 2

Utilizar GDB para corregir el programa.

NOTA: Para compilar el código de la actividad, ejecutar:

**\$ gcc -w actividad2.c -o actividad2 -lm**

```

1 #include <stdio.h>
2 #include <math.h>
3
4 void main()
5 {
6     int K, AP, N;
7     double X, AS;
8     printf("Ingresa cuántos términos calcular de la serie: X^K/K!");
9     printf("\nN=");
10    scanf("%i",&N);
11    printf("X=");
12    scanf("%lf",&X);
13    K=0;
14    AP=1;
15    AS=0;
16    while(K<=N)
17    {
18        AS=AS+pow(X,K)/AP;
19        K=K+1;
20        AP=AP*K;
21    }
22    printf("Resultado=%le",AS);
23 }
24

```

```

GNU gdb (Ubuntu 8.1-0ubuntu3.1) 8.1-0.20180409-git
Copyright (C) 2018 Free Software Foundation, Inc.
License GPLv3+: GNU GPL version 3 or later <http://gnu.org/licenses/gpl.html>
This is free software: you are free to change and redistribute it.
There is NO WARRANTY, to the extent permitted by law. Type "show copying"
and "show warranty" for details.
This GDB was configured as "x86_64-linux-gnu".
Type "show configuration" for configuration details.
Para las instrucciones de informe de errores, vea:
<http://www.gnu.org/software/gdb/bugs/>.
Find the GDB manual and other documentation resources online at:
<http://www.gnu.org/software/gdb/documentation/>.
For help, type "help".
Type "apropos word" to search for commands related to "word"...
Leyendo símbolos desde ./act2... (no se encontraron símbolos de depuración) hecho.
(gdb) run
Starting program: /home/julio/Descargas/act2
Ingresa cuántos términos calcular de la serie: X^K/K!
N= 3
X=5
Resultado=3.93333e+01[Inferior 1 (process 2176) exited normally]
(gdb) _

```

```

jullo@jullo-VirtualBox ~/Descargas $ gcc -w actividad2.c -o actividad2 -lm
jullo@jullo-VirtualBox ~/Descargas $ ./actividad2
Ingrese cuántos términos calcular de la serie: X^K/K!
N=3
Violación de segmento
jullo@jullo-VirtualBox ~/Descargas $ _

```

### Actividad 3

Utilizar GDB para corregir el programa

```

1  #include <stdio.h>
2
3  int main()
4  {
5      int numero;
6
7      printf("Ingrese un número:\n");
8      scanf("%i",&numero);
9
10     long int resultado = 1;
11     while(numero>=0){
12         numero--;
13         resultado *= numero;
14     }
15
16     printf("El factorial de %i es %li.\n", numero, resultado);
17
18     return 0;
19 }
20
21

```

```

jullo@jullo-VirtualBox ~/Descargas $ gcc -g actividad3.c -o actividad3
jullo@jullo-VirtualBox ~/Descargas $ gdb ./actividad3
GNU gdb (Ubuntu 8.1-0ubuntu3.1) 8.1.0.20180409-git
Copyright (C) 2018 Free Software Foundation, Inc.
License GPLv3+: GNU GPL version 3 or later <http://gnu.org/licenses/gpl.html>
This is free software: you are free to change and redistribute it.
There is NO WARRANTY, to the extent permitted by law. Type "show copying"
and "show warranty" for details.
This GDB was configured as "x86_64-linux-gnu".
Type "show configuration" for configuration details.
Para las instrucciones de informe de errores, vea:
<http://www.gnu.org/software/gdb/bugs/>.
Find the GDB manual and other documentation resources online at:
<http://www.gnu.org/software/gdb/documentation/>.
For help, type "help".
Type "apropos word" to search for commands related to "word"...
Leyendo símbolos desde ./actividad3...hecho.
(gdb) run
Starting program: /home/jullo/Descargas/actividad3
Ingrese un número: 3
El factorial de -1 es 0.
[Inferior 1 (process 2217) exited normally]
(gdb) _

```



```
Peppermint Terminal
actividad3.c
B+> 4 {
5     int numero;
6
7     printf("Ingrese un número:\n");
8     scanf("%i",&numero);
9
10    long int resultado = 1;
11    while(numero>=0){
12        numero--;
13        resultado *= numero;
14    }
15    printf("El factorial de %i es %li.\n", numero, resultado);
16
17    return 0;
18 }

native process 2266 In: main L4 PC: 0x555555554772
(gdb) start
Punto de interrupción temporal 1 at 0x555555554772: file actividad3.c, line 4.
Starting program: /home/julio/Descargas/actividad3
Temporary breakpoint 1, main () at actividad3.c:4
(gdb) _
```

```
julio@julio-VirtualBox ~/Descargas $ ./actividad1
Ingresa un número: 1

El resultado es: 1
julio@julio-VirtualBox ~/Descargas $ ./actividad1
Ingresa un número: 2

El resultado es: 1
julio@julio-VirtualBox ~/Descargas $ ./actividad1
Ingresa un número: 3

El resultado es: 4
julio@julio-VirtualBox ~/Descargas $ ./actividad1
Ingresa un número: 4

El resultado es: 4
julio@julio-VirtualBox ~/Descargas $ ./actividad1
Ingresa un número: 5

El resultado es: 9
julio@julio-VirtualBox ~/Descargas $ ./actividad1
Ingresa un número: 6

El resultado es: 9
julio@julio-VirtualBox ~/Descargas $ ./actividad1
Ingresa un número: 7

El resultado es: 16
julio@julio-VirtualBox ~/Descargas $ ./actividad1
Ingresa un número: 8

El resultado es: 16
```

## CONCLUSION

En esta práctica identificamos los datos de un programa que no dejan que este se efectúe de manera asertiva, para esto utilizamos GDB que en esta opción mac no tiene respuesta, corrimos la lista, identificamos la parte errónea y le dimos solución.  
nota: Utilizo mac para resolver mis prácticas.