	<b>Carátula para entrega de prácticas</b>	
Facultad de Ingeniería		Laboratorio de docencia

# Laboratorios de computación salas A y B

<i>Profesor:</i>	M.C. Alejandro Esteban Pimentel Alarcón
<i>Asignatura:</i>	Fundamentos de Programación
<i>Grupo:</i>	3
<i>No de Práctica(s):</i>	10
<i>Integrante(s):</i>	Cárdenas Belmont Alan- 5783 Sotelo Leyva Carlos - 2751
<i>No. de Equipo de cómputo empleado:</i>	23
<i>No. de Lista o Brigada:</i>	07 y 47
<i>Semestre:</i>	2020-1
<i>Fecha de entrega:</i>	28/10/19
<i>Observaciones:</i>	No se cumple el objetivo de la actividad 1. No pusieron descripción alguna

**CALIFICACIÓN:** 8

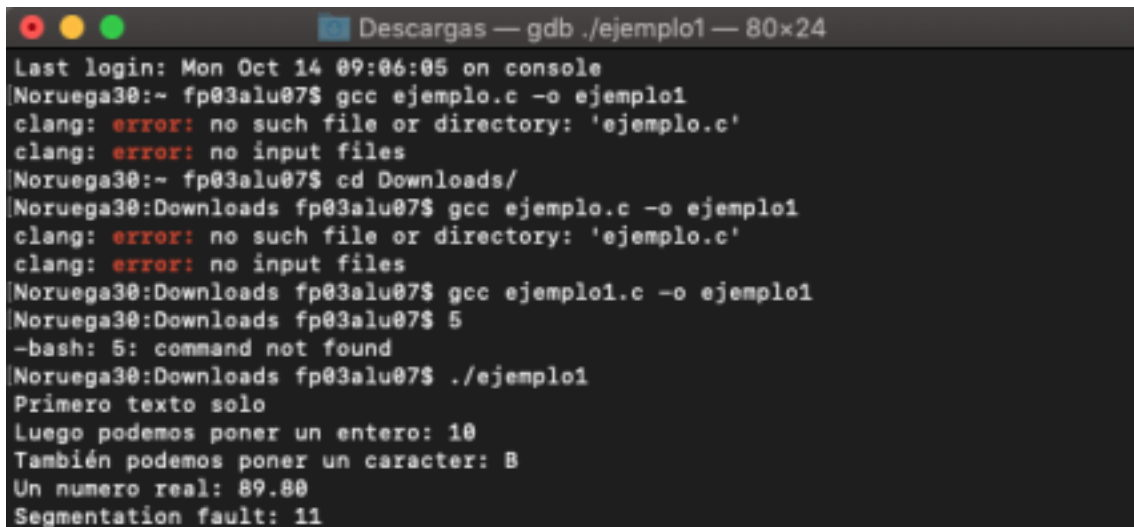
# Introducción

Esta práctica se basa principalmente en la corrección de errores en el programa por medio de la implementación de nuevos comandos de revisión, y depuración del lenguaje C y así corregir errores y posibles errores.

## Objetivo

Aprender las técnicas básicas de depuración de programas en C para revisar de manera precisa el flujo de ejecución de un programa y el valor de las variables; en su caso, corregir posibles errores.

## Ejemplo

A terminal window titled 'Descargas — gdb ./ejemplo1 — 80x24'. The terminal shows a user named 'Noruega30' at a machine 'fp03alu07\$'. They attempt to compile 'ejemplo.c' with 'gcc', but receive 'clang: error: no such file or directory: 'ejemplo.c'' and 'clang: error: no input files'. They then move to the 'Downloads' directory and repeat the compilation, still receiving the same errors. They then compile 'ejemplo1.c' with 'gcc', which succeeds. They run the program with '5' as input, but get '-bash: 5: command not found'. Finally, they run './ejemplo1', which outputs: 'Primero texto solo', 'Luego podemos poner un entero: 10', 'También podemos poner un caracter: B', 'Un numero real: 89.80', and 'Segmentation fault: 11'.

```
Descargas — gdb ./ejemplo1 — 80x24
Last login: Mon Oct 14 09:06:05 on console
Noruega30:~ fp03alu07$ gcc ejemplo.c -o ejemplo1
clang: error: no such file or directory: 'ejemplo.c'
clang: error: no input files
Noruega30:~ fp03alu07$ cd Downloads/
Noruega30:Downloads fp03alu07$ gcc ejemplo.c -o ejemplo1
clang: error: no such file or directory: 'ejemplo.c'
clang: error: no input files
Noruega30:Downloads fp03alu07$ gcc ejemplo1.c -o ejemplo1
Noruega30:Downloads fp03alu07$ 5
-bash: 5: command not found
Noruega30:Downloads fp03alu07$ ./ejemplo1
Primero texto solo
Luego podemos poner un entero: 10
También podemos poner un caracter: B
Un numero real: 89.80
Segmentation fault: 11
```

```
Descargas — gdb ./ejemplo1 — 80x24
Noruega30:Downloads fp03alu07$ gcc -g ejemplo1.c -o ejemplo1
Noruega30:Downloads fp03alu07$ gdb ./ejemplo1
GNU gdb (GDB) 8.2
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-apple-darwin18.2.0".
Type "show configuration" for configuration details.
For bug reporting instructions, please see:
<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"...
Reading symbols from ./ejemplo1...Reading symbols from /Users/fp03alu07/Download
s/ejemplo1.dSYM/Contents/Resources/DWARF/ejemplo1...done.
done.
```

```
fp03alu07 — fp03alu07@samba:~ — ssh fp03alu07@192.168.2.200 — 80x24
Last login: Mon Oct 14 09:18:40 on ttys000
Noruega30:~ fp03alu07$ servidor
Noruega30:~ fp03alu07$ ssh fp03alu07@192.168.2.200
The authenticity of host '192.168.2.200 (192.168.2.200)' can't be established.
RSA key fingerprint is SHA256:jTgFsbvP7IaIpwchV27DaUa9i2pvAVVZwZzbIneOF8.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '192.168.2.200' (RSA) to the list of known hosts.
fp03alu07@192.168.2.200's password:
Samba

-bash: aviso: setlocale: LC_CTYPE: no se puede cambiar el local (UTF-8)
[fp03alu07@samba ~]$
```



La depuración de un programa es útil cuando:

Se desea optimizar el programa, no basta que el programa se pueda compilar y se someta a pruebas que demuestren que funciona correctamente. Debe realizarse un análisis exhaustivo del mismo en ejecución para averiguar cuál es su flujo de operación y encontrar formas de mejorarlo (reducir el código, utilizar menos recursos llegando a los mismos resultados, hacer menos rebuscado al algoritmo), o bien, encontrar puntos donde puede fallar con ciertos tipos de entrada de datos.

Ejecutar el programa: se procede a ejecutar el programa en la herramienta de depuración ofreciendo diversas opciones para ello.  $\infty$  Mostrar el código fuente del programa: muestra cuál fue el código fuente del programa con el número de línea con el fin de emular la ejecución del programa sobre éste, es decir, se indica qué parte del código fuente se está ejecutando a la hora de correr el programa.

Ejecutar la siguiente instrucción: cuando la ejecución del programa se ha detenido por medio del depurador, esta función permite ejecutar una instrucción más y detener el programa de nuevo. Esto es útil cuando se desea estudiar detalladamente una pequeña sección del programa. Si en la ejecución existe una llamada a función se ingresará a ella.

Ejecutar la siguiente línea: es muy similar a la función anterior, pero realizará todas las instrucciones necesarias hasta llegar a la siguiente línea de código. Si en la ejecución existe una llamada a función se ignorará.

# Actividad 1

```
vanessa@Titan: ~/Escritorio
Archivo Editar Ver Buscar Terminal Ayuda
9 actividad1.c FP_2020-1_8166 promedio.c
vanessa@Titan:~/Escritorio$ gcc -g actividad1.c -o act1
vanessa@Titan:~/Escritorio$ gdb ./act1
GNU gdb (Ubuntu 8.2-0ubuntu1) 8.2
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.
For bug reporting instructions, please see:
<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"...
Legendo símbolos desde ./act1...hecho.
(gdb) run
Starting program: /home/vanessa/Escritorio/act1
Ingresa un número: 1

El resultado es: 1
[Inferior 1 (process 3185) exited normally]
(gdb) list
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)
(gdb) █
```

```
vanessa@Titan: ~/Escritorio
Archivo Editar Ver Buscar Terminal Ayuda
actividad1.c
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     }

native process 3204 In: main L9 PC: 0x55555555518b
(gdb) start
Punto de interrupción temporal 1 at 0x55555555515d: file actividad1.c, line 4.
Starting program: /home/vanessa/Escritorio/act1

Temporary breakpoint 1, main () at actividad1.c:4
(gdb) n
Ingresa un número: 3
```

Este fue el resultado después de correrlo en la terminal:

```
lades  Terminal  mar, 22 de oct, 00:18 ●
vanessa@Titan: ~/Escritorio

Archivo  Editar  Ver  Buscar  Terminal  Ayuda

vanessa@Titan:~$ cd Escritorio
vanessa@Titan:~/Escritorio$ ls
1  a2      actividad1.c      FP_2020-1_8166  promedio.c
2  a3      actividad2.c      main            'tips para el ensayo.pptx'
3  ac2     actividad2.c      primo
8  act1    actividad3.c      primo.c
9  act3    ensayorúbrica.docx  promedio
vanessa@Titan:~/Escritorio$ gcc actividad1.c -o ac1
vanessa@Titan:~/Escritorio$ ./ac1
Ingresa un número: 4

El resultado es: 4
vanessa@Titan:~/Escritorio$ ./ac1
Ingresa un número: 5

El resultado es: 9
vanessa@Titan:~/Escritorio$ ./ac1
Ingresa un número: 6

El resultado es: 9
vanessa@Titan:~/Escritorio$ ./ac1
Ingresa un número: 7

El resultado es: 16
vanessa@Titan:~/Escritorio$ ./ac1
Ingresa un número: 2

El resultado es: 1
vanessa@Titan:~/Escritorio$ ./ac1
Ingresa un número: 9

El resultado es: 25
vanessa@Titan:~/Escritorio$
```

No hay descripción



## Actividad 2

```
GNU gdb (Ubuntu 8.2-0ubuntu1) 8.2
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.
For bug reporting instructions, please see:
<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 ./a2...hecho.
(gdb) run
Starting program: /home/vanessa/Escritorio/a2
Ingrese cuántos términos calcular de la serie: X^K/K!
N=4
X=5
Resultado=6.537500e+01[Inferior 1 (process 2427) exited normally]
(gdb) list
1      #include <stdio.h>
2      #include <math.h>
3
4      void main()
5      {
6          int K, AP, N;
7          double X, AS;
8          printf("Ingrese cuántos términos calcular de la serie: X^K/K!");
9          printf("\nN=");
10         scanf("%i",&N);
(gdb) █
```

```
actividad2.c
5      {
6          int K, AP, N;
7          double X, AS;
8          printf("Ingrese 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
25
26

native process 2380 In: main
errp=errp@entry=0x0 at vfscanf.c:1881
(gdb) start
The program being debugged has been started already.
Start it from the beginning? (y or n) y
Punto de interrupción temporal 2 at 0x5555555515d: file actividad2.c, line 8.
Starting program: /home/vanessa/Escritorio/ac2

Temporary breakpoint 2, main () at actividad2.c:8
(gdb) n
(gdb) n
(gdb) n
N=
```



```

1  #include <stdio.h>
2  #include <math.h>
3
4  void main()
5  {
6      int K, AP, N;
7      double X, AS;
8      printf("Ingrese 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\n",AS);
23 }
24

```

### Actividad 3

```

GNU gdb (Ubuntu 8.2-0ubuntu1) 8.2
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.
For bug reporting instructions, please see:
<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 ./act3...hecho.
(gdb) break 11
Punto de interrupción 1 at 0x11a8: file actividad3.c, line 11.
(gdb) run
Starting program: /home/vanessa/Escritorio/act3
Ingrese un número:
6

Breakpoint 1, main () at actividad3.c:11
11         while(numero>=0){
(gdb)

```

```
actividad3.c
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
22
23
24
25

native process 2261 In: main
(gdb) start
The program being debugged has been started already.
Start it from the beginning? (y or n) y
Punto de interrupción temporal 2 at 0x5555555510d: file actividad3.c, line 4.
Starting program: /home/vanessa/Escritorio/act3

Temporary breakpoint 2, main () at actividad3.c:4
(gdb) print numero
$1 = 21845
(gdb) display numero
1: numero = 21845
(gdb)
```

```
Ingrese un número:
3
El factorial de 3 es 6
vanessa@Titan:~/Escritorio$ ./ac3
Ingrese un número:
5
El factorial de 5 es 120
vanessa@Titan:~/Escritorio$ ./ac3
Ingrese un número:
26
El factorial de 26 es -1569523520172457984
vanessa@Titan:~/Escritorio$ ./ac3
Ingrese un número:
16
El factorial de 16 es 20922789888000
vanessa@Titan:~/Escritorio$
```

## Conclusión

Esta práctica me parece muy útil a futuros programas, ya que nos ahorra mucho tiempo en la detección de errores en el mismo, es importante usar gdb en programas en lenguaje C.