



Carátula para entrega de prácticas

Facultad de Ingeniería

Laboratorio de docencia

Laboratorios de computación salas A y B

Alejandro Pimentel Alarco

Profesor:

Fundamentos de Programación

Asignatura:

3

Grupo:

10

No de Práctica(s):

Badillo Ruiz Evangelina

Integrante(s):

*No. de Equipo de
cómputo empleado:* 36
3069

No. de Lista o Brigada:

1

Semestre:

21 de Octubre del 2019

Fecha de entrega:

Observaciones: En general bien, pero en la última actividad no se ve el programa corregido funcionando, y el la primera se tenía que describir el programa nada más, no había nada que corregir.

CALIFICACIÓN: 8

PRACTICA 10

BADILLO RUIZ EVANGELINA NO.CUENTA 3069, NL.4



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.

INTRODUCCIÓN

- Aprenderemos usar la terminal de gdb, dentro de la terminal de Linux
- Para eso aprenderemos usar el código gcc -g _____ -o _____
- Para entrar a correrlo en la terminal gdb ./_____
- Utilizaremos los comandos run , list(l), quit (q), CTRL+x+a, start, next (n), CTRL+l, break (b), print (p), display.Para la terminal gdb
- Utilizaremos el comando gcc -w _____ -o _____ -lm
- Gdb nos ayuda a saber dentro de la terminal de Linux donde tenemos errores en el código.

gcc -g _____ -o _____

**Sirve para compilar el
archivo primero
Que esta como texto plano.
Para que entre en la terminal
de gdb**

LINUX [Conrado] - Oracle VM VirtualBox

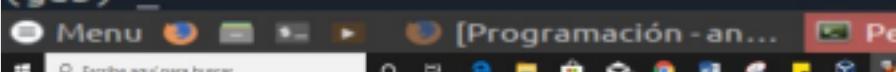
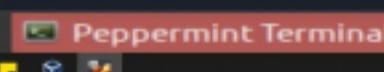
Peppermint Terminal

```
evangelina@evangelina-VirtualBox ~ $ cd Descargas/
evangelina@evangelina-VirtualBox ~/Descargas $ gcc -g act1.c -o act1
evangelina@evangelina-VirtualBox ~/Descargas $ gdb ./act1
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 ./act1...hecho.
(gdb) run
Starting program: /home/evangelina/Descargas/act1
Ingresá un número:
5
El resultado es: 9
[Inferior 1 (process 1607) exited normally]
(gdb) _
```

gdb ./_____

Sirve para que a partir de la terminal de Linux entremos, a la terminal de gdb.

```
evangelina@evangelina-VirtualBox ~/Descargas $ gdb./ejemplo1
bash: gdb./ejemplo1: No existe el archivo o el directorio
evangelina@evangelina-VirtualBox ~/Descargas $ gdb ./ejemplo1
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 ./ejemplo1...hecho.
(gdb) _
```

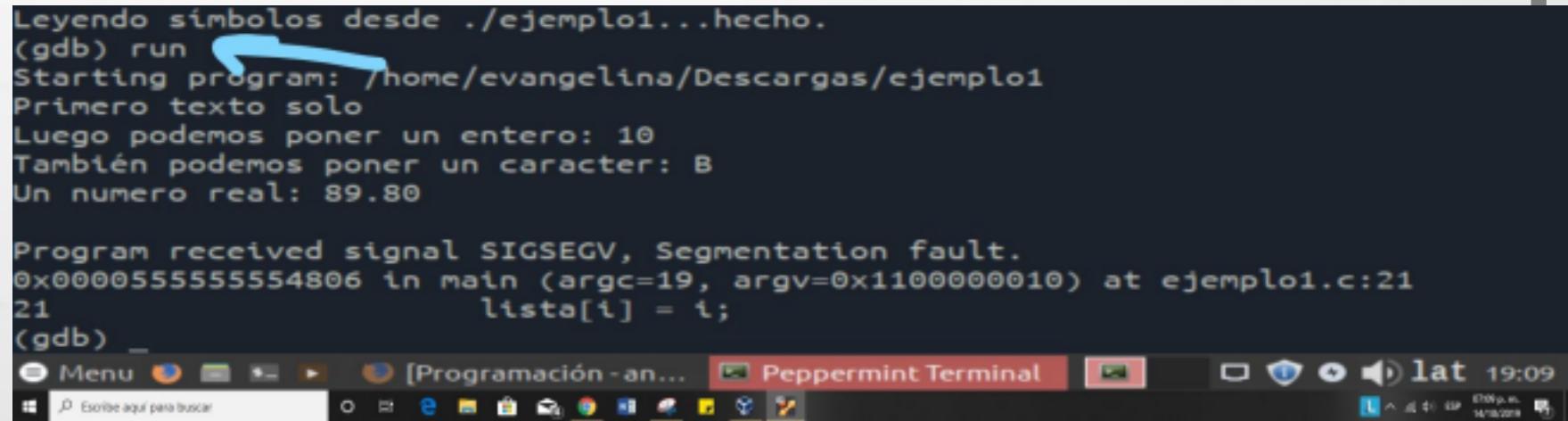


run

**Sirve para que dentro de la terminal de gdb lo corra.
Ademas te dice la linea donde hay error**

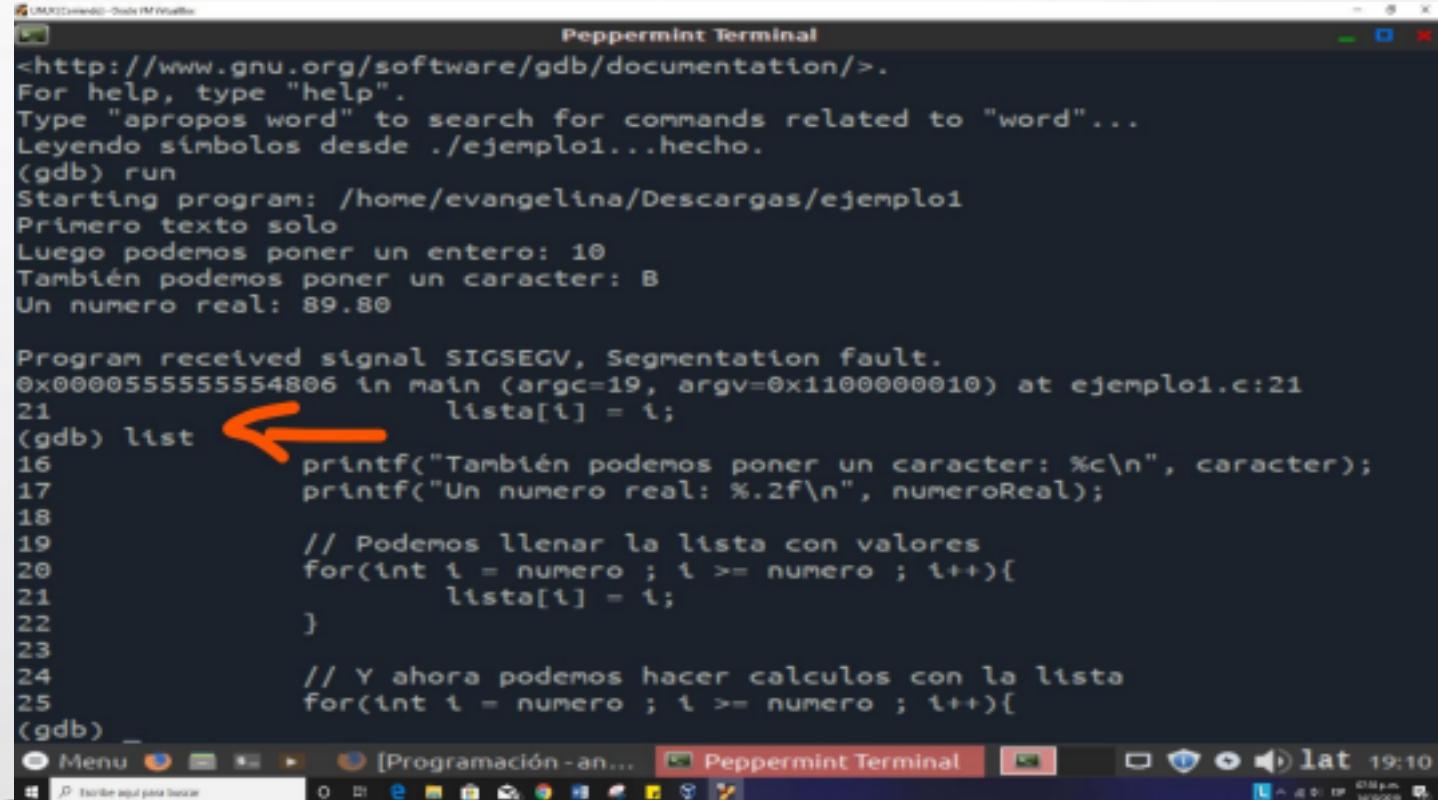
```
Leyendo símbolos desde ./ejemplo1...hecho.
(gdb) run
Starting program: /home/evangelina/Descargas/ejemplo1
Primero texto solo
Luego podemos poner un entero: 10
También podemos poner un carácter: B
Un número real: 89.80

Program received signal SIGSEGV, Segmentation fault.
0x000055555554806 in main (argc=19, argv=0x1100000010) at ejemplo1.c:21
21                      lista[i] = i;
(gdb) _
```



list()

Nos enlista dentro de la terminal de Linux el programa.

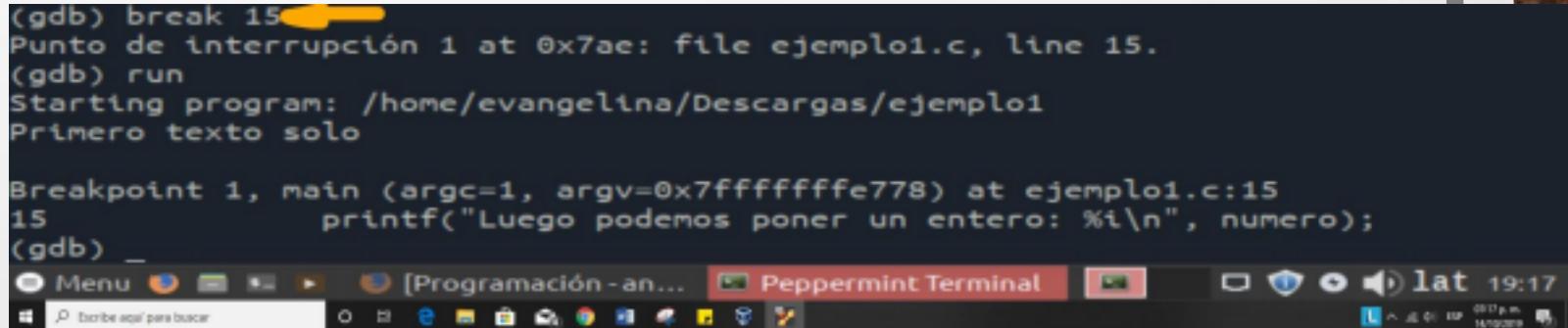


```
Peppermint Terminal
<http://www.gnu.org/software/gdb/documentation/>.
For help, type "help".
Type "apropos word" to search for commands related to "word"...
Leyendo símbolos desde ./ejemplo1...hecho.
(gdb) run
Starting program: /home/evangelina/Descargas/ejemplo1
Primero texto solo
Luego podemos poner un entero: 10
También podemos poner un carácter: B
Un número real: 89.80

Program received signal SIGSEGV, Segmentation fault.
0x0000555555554806 in main (argc=19, argv=0x1100000010) at ejemplo1.c:21
21      lista[i] = i;
(gdb) list
16      printf("También podemos poner un carácter: %c\n", carácter);
17      printf("Un número 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 cálculos con la lista
25      for(int i = numero ; i >= numero ; i++){
(gdb) _
```

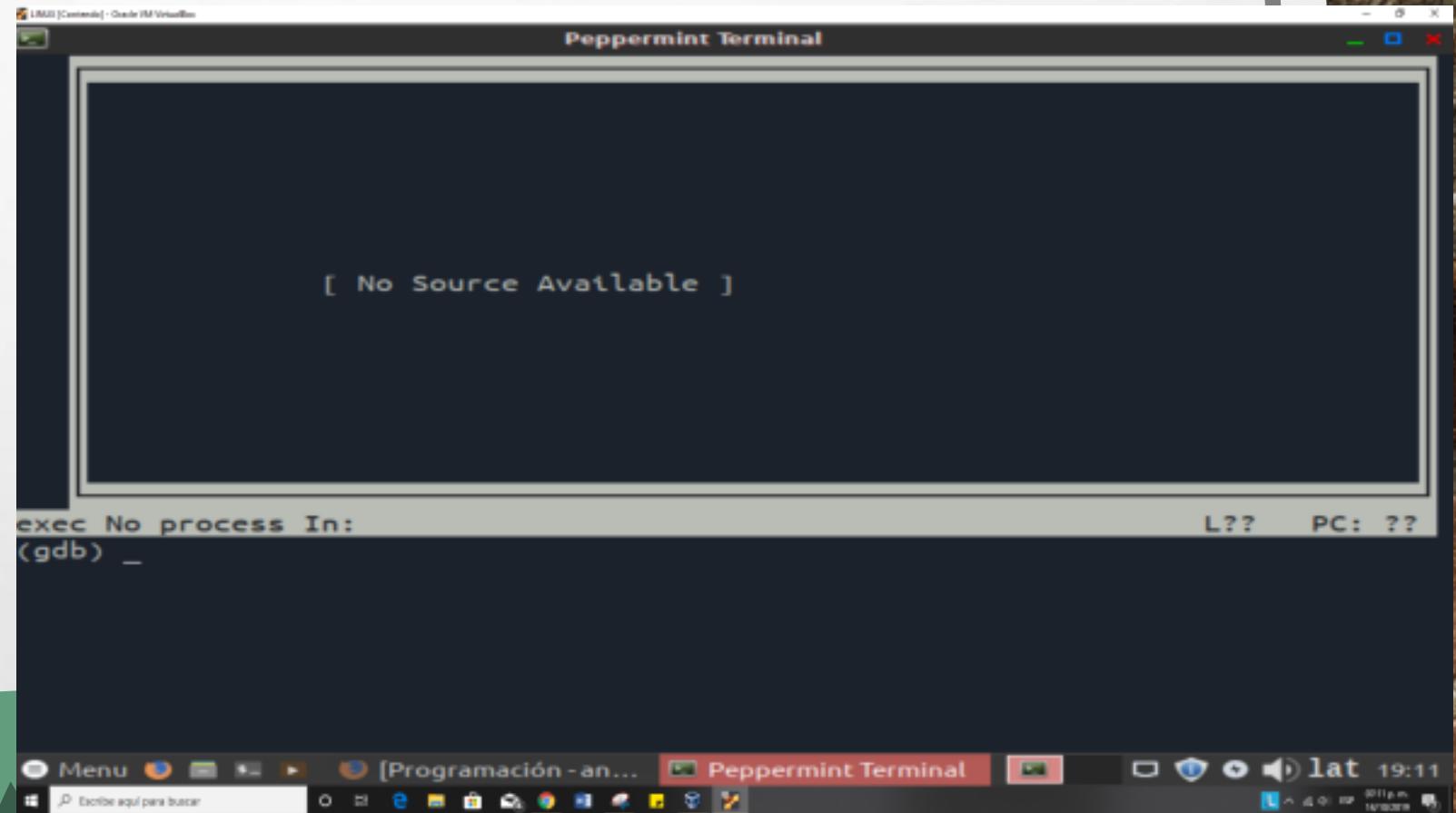
break (b)

**Le indicamos a partir de
donde queremos que
termine el programa de
correr**



CTRL+x+a

**Es para entrar a la ver el
error en el programa a partir
de la terminal gdb.**



start

Es para que comience el
programa a mostrarte línea
por línea el código.

The screenshot shows a Linux desktop environment with a terminal window titled "Peppermint Terminal". The terminal displays the following content:

```
ejemplo1.c
B+> 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
14            printf("Primero texto solo\n");
15            printf("Luego podemos poner un entero: %i\n", numero);
16            printf("También podemos poner un carácter: %c\n", caracter);
17            printf("Un número real: %.2f\n", numeroReal);

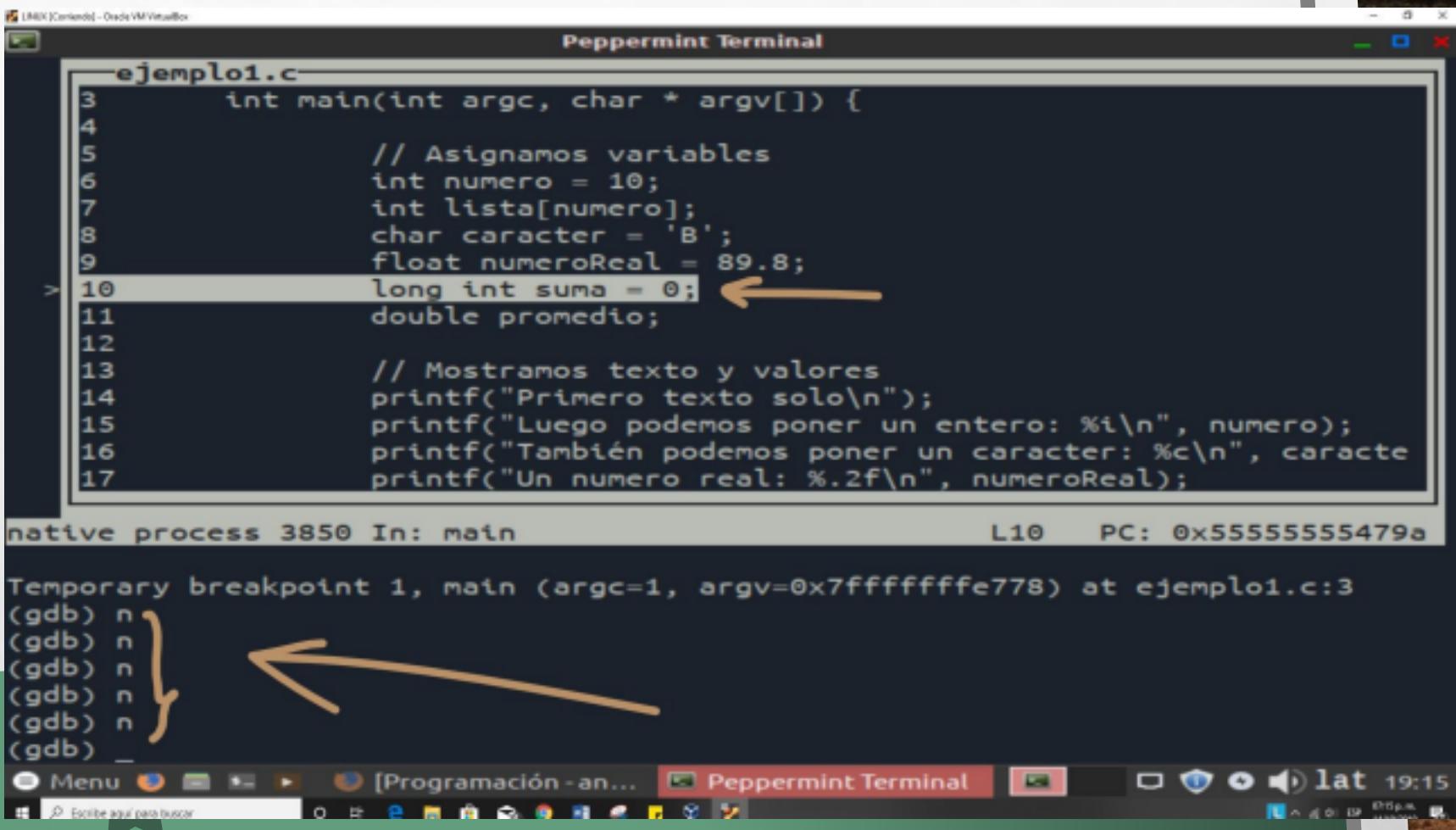
native process 3850 In: main
(gdb) start
Punto de interrupción temporal 1 en 0x70a: archivo ejemplo1.c, línea 3.
Starting program: /home/evangelina/Descargas/ejemplo1

Temporary breakpoint 1, main (argc=1, argv=0x7fffffff778) en ejemplo1.c:3
(gdb) _
```

The terminal window is part of a desktop environment with a menu bar, a search bar, and a taskbar at the bottom.

next (n)

Next sirve para ir bajando de
línea en línea.Una vez acabe
el programa hay que volver a
escribir Start



Peppermint Terminal

```
ejemplo1.c
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
14        printf("Primero texto solo\n");
15        printf("Luego podemos poner un entero: %i\n", numero);
16        printf("También podemos poner un carácter: %c\n", caracte
17        printf("Un numero real: %.2f\n", numeroReal);
```

native process 3850 In: main L10 PC: 0x55555555479a

Temporary breakpoint 1, main (argc=1, argv=0x7fffffff778) at ejemplo1.c:3

(gdb) n } ←

(gdb) n

(gdb) n

(gdb) n

(gdb) n

(gdb) n

(gdb) _

Menu [Programación - an... Peppermint Terminal lat 19:15

Escríbelo aquí para buscar

CTRL+I

**Es cuando comienza a tener
error el programa, y no esta
trabajando adecuadamente.**

CTRL+L

Peppermint Terminal

```
ejemplo1.c
9         float numeroReal = 89.8;
10        long int suma = 0;
11        double promedio;
12
13        // Mostramos texto y valores
14        printf("Primero texto solo\n");
15        printf("Luego podemos poner un entero: %i\n", numero);
16        printf("TambiM-CM-)n podemos poner un caracter: %c\n", caract
>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;

native process 22254 In: main                                         L17   PC: 0x40006b1
(gdb) start
Punto de interrupci n temporal 1 at 0x4005e6: file ejemplo1.c, line 3.
Starting program: /home/alejandro/ejemplo1

Temporary breakpoint 1, main (argc=1, argv=0xfffffffffe748) at ejemplo1.c:3
(gdb) next
(gdb) n
(gdb) _
```

quit (q)

Es para salir del programa de gdb, para regresar a la terminal de linux

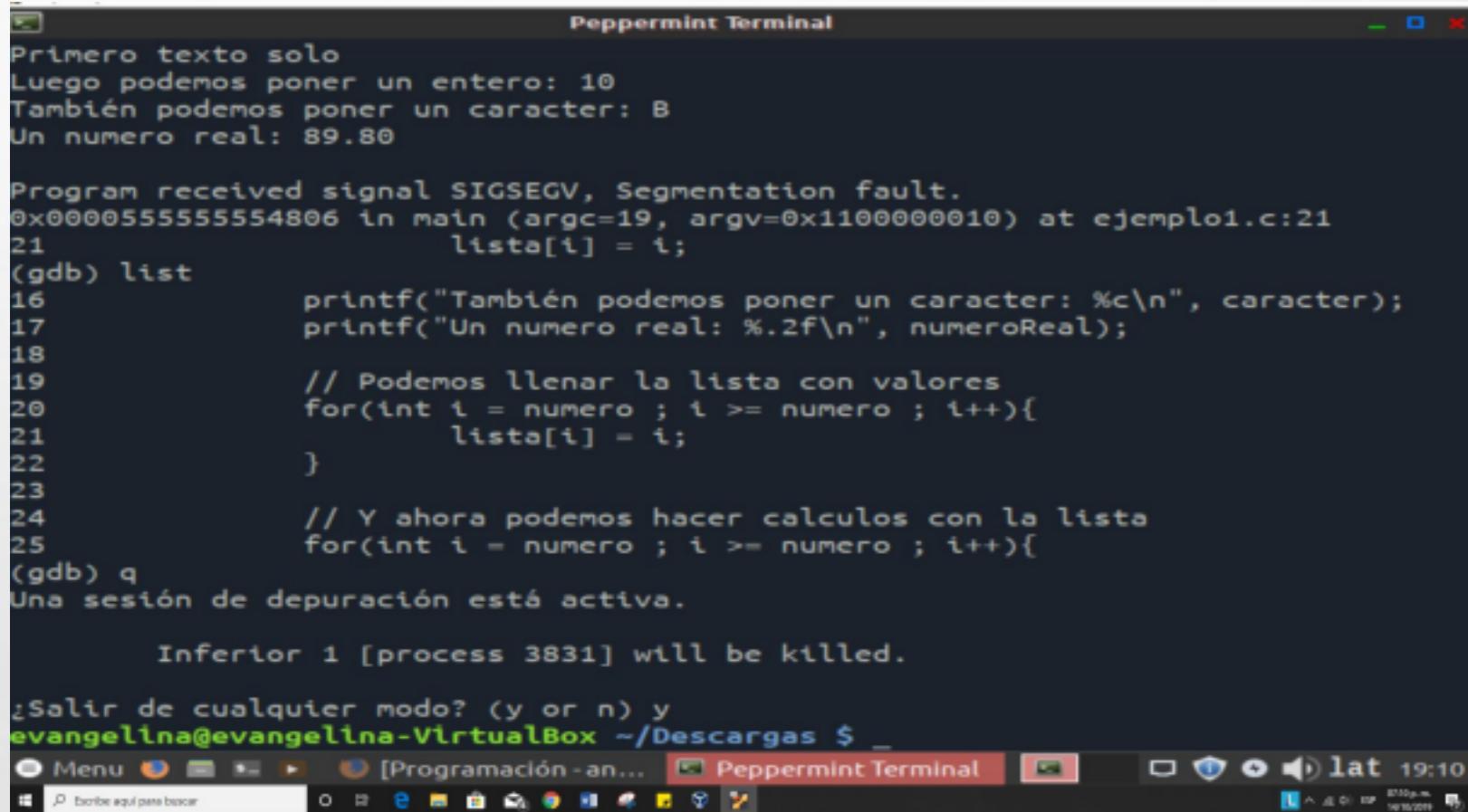
```
Peppermint Terminal

Primer texto solo
Luego podemos poner un entero: 10
También podemos poner un caracter: B
Un numero real: 89.80

Program received signal SIGSEGV, Segmentation fault.
0x0000555555554806 in main (argc=19, argv=0x1100000010) at ejemplo1.c:21
21             lista[i] = i;
(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) q
Una sesión de depuración está activa.

Inferior 1 [process 3831] will be killed.

¿Salir de cualquier modo? (y or n) y
evangelina@evangelina-VirtualBox ~/Descargas $ _
```



`print(p)`

Es para saber los valores que tiene asignadas las variables.

```
LINUX [CentOS] - Distro VM VirtualBox  
Peppermint Terminal  
ejemplo1.c  
9         float numeroReal = 89.8;  
10        long int suma = 0;  
11        double promedio;  
12  
13        // Mostramos texto y valores  
14        printf("Primero texto solo\n");  
15        printf("Luego podemos poner un entero: %i\n", numero);  
16        printf("También podemos poner un caracter: %c\n", caracte  
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  
native process 3913 In: main  
(gdb) n  
(gdb) print i  
$1 = 11  
(gdb) _
```

display

Es para saber los valores que tiene asignadas las variables.Especificamente

The screenshot shows a Linux desktop environment with a terminal window titled "Peppermint Terminal". The terminal displays the following content:

```
ejemplo1.c
9     float numeroReal = 89.8;
10    long int suma = 0;
11    double promedio;
12
13    // Mostramos texto y valores
14    printf("Primero texto solo\n");
15    printf("Luego podemos poner un entero: %i\n", numero);
16    printf("También podemos poner un caracter: %c\n", caracte
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

native process 3913 In: main
(gdb) n
(gdb) print i
$1 = 11
(gdb) display i
1: i = 11
(gdb)
```

The terminal window is part of a desktop environment with a menu bar, a search bar, and a taskbar at the bottom.

Act.1

```
#include <stdio.h>

void main()
{
    int N, CONT, AS;
    AS=0;
    CONT=1;
    printf("Ingresa un número: ");
    scanf("%i", &N);
    while(CONT<=N)
    {
        AS=(AS+CONT);
        CONT=(CONT+2);
    }
    printf("\nEl resultado es: %i\n", AS);
}
```

gdb

```
evangelina@evangelina-VirtualBox ~/Descargas $ gdb ./act1
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 ./act1...(no se encontraron símbolos de depuración)hecho
.
(gdb) run
Starting program: /home/evangelina/Descargas/act1
Ingresa un número: 5
```

```
(gdb) run
Starting program: /home/evangelina/Descargas/act1
Ingresa un número: 5

El resultado es: 9
[Inferior 1 (process 11407) exited normally]
(gdb)
(gdb) ls
orden indefinida: "ls". Intente con «help»
(gdb) list
1      <built-in>: No existe el archivo o el directorio.
(gdb) run
Starting program: /home/evangelina/Descargas/act1
Ingresa un número: 5

El resultado es: 9
[Inferior 1 (process 11415) exited normally]
(gdb) list
1      <built-in>: No existe el archivo o el directorio.
evangelina@evangelina-VirtualBox ~/Descargas $
```

Run y list

start

Ubuntu 18.04 LTS - 64 bits - 2018-04-09 - Peppermint Terminal

```
act1.c
B+> 4
5      int N, CONT, AS;
6      AS=0;
7      CONT=1;
8      printf("Ingresa un número:\n");
9      scanf("%i", &N);
10     while(CONT<=N)
11     {
12         AS=(AS+CONT);
13         CONT=(CONT+2);
14     }
15     printf("El resultado es: %i\n", AS);
16
17
18
```

Native process 1616 In: main L4 PC: 0x555555554772
(gdb) start
Punto de interrupción temporal 1 en 0x555555554772: archivo act1.c, línea 4.
Starting program: /home/evangelina/Descargas/act1
Temporary breakpoint 1, main () en act1.c:4
(gdb) _

Correccion act1



```
#include <stdio.h>

void main()
{
    int N, CONT, AS;
    AS=0;
    CONT=1;
    printf("Ingresa un número:\n");
    scanf("%i",&N);
    while(CONT<=N)
    {
        AS=(AS+CONT);
        CONT=(CONT+2);
    }
    printf("El resultado es: %i\n", AS);
}
```

No había errores en la actividad 1. Solo se tenía que usar GDB para describir el funcionamiento del programa

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

void main()
{
    int K, AP, N;
    double X, AS;
    printf("Ingrese cuántos términos calcular de la serie: X^K/K!");
    printf("\nN=");
    scanf("%i", &N);
    printf("X=");
    scanf("%lf", &X);
    K=0;
    AP=1;
    AS=0;
    while(K<=N)
    {
        AS=AS+pow(X,K)/AP;
        K=K+1;
        AP=AP*K;
    }
    printf("Resultado=%le", AS);
}
```

Run y list

```
evangelina@evangelina-VirtualBox ~/Descargas $ gcc -w act2.c -o act2
evangelina@evangelina-VirtualBox ~/Descargas $ gdb ./act2
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/evangelina/Descargas/act2
Ingrese cuántos términos calcular de la serie: X^K/K;
N=5

X=6
Resultado=1.900833e+01[Inferior 1 (process 2307) exited normally]
(gdb) __
● Menu ○ Peppermint Terminal lat 01:10
```

gdb

ACT.2 GCC -W_____ -O_____ -LM



gdb

```
evangelina@evangelina-VirtualBox ~/fp_2020-1_3069 $ cd /home/evangelina/Descargas  
as/  
evangelina@evangelina-VirtualBox ~/Descargas $ ls  
act1      main          operadores.c      tareas3  
act1.c    main4         operadoreslogicos  tiposdevariables-i  
act2.c    main.c        operadoreslogicos.c tiposdevariables-i.c  
act3.c    mostraryleer   prueba1  
ejemplo1  mostraryleer.c tareas1  
ejemplo1.c operadores   tareas2  
evangelina@evangelina-VirtualBox ~/Descargas $ gcc -g act2.c -o act2  
evangelina@evangelina-VirtualBox ~/Descargas $ gdb ./act2  
GNU gdb (Ubuntu 8.1.0-ubuntu3.1) 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"....  
Peppermint Terminal
```

list

```
(gdb) l  
1  pri #include <stdio.h>  
2  sec #include <math.h>  
3  K= double pow(double x,double y){  
4  AP= double suma;  
5  AS=0; suma=x+y;  
6  while( return suma;  
7  }  
8  void main()  
9  {  
10     int K, AP, N;  
(gdb) _  
Peppermint Terminal
```

Run

```
evangelina@evangelina-VirtualBox ~/Descargas $ gdb ./activida2  
GNU gdb (Ubuntu 8.1.0-ubuntu3.1) 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 ./activida2...hecho.  
(gdb) run  
Starting program: /home/evangelina/Descargas/activida2  
Ingrese cuántos términos calcular de la serie: X^K/K!  
N=5  
AS=AS+pow(X,K)/AP;  
K=K+1;  
X=5  
Resultado=1.629167e+01[Inferior 1 (process 10147) exited normally]  
(gdb) 3  
Peppermint Terminal
```

start

```
act2.c  
B+> 9  {  
10     int K, AP, N;  
11     double X, AS;  
12     printf("\nN=");  
13     scanf("%t",& N);  
14     printf("\nX=");  
15     scanf("%lf",& X);  
16     K=0;  
17     AP=1;  
18     AS=0;  
19     while(K<=N)  
20     {  
21         AS=AS+pow(X,K)/AP;  
22         K=K+1;  
23     }  
native process 2682 In: main  
(gdb) start  
Punto de interrupción temporal 1 at 0x555555554746: file act2.c, line 9.  
Starting program: /home/evangelina/Descargas/act2  
Temporary breakpoint 1, main () at act2.c:9  
(gdb) _  
Peppermint Terminal
```

Act.3

```
#include <stdio.h>

int main()
{
    int numero;

    printf("Ingrese un número:\n");
    scanf("%i", &numero);

    long int resultado = 1;
    while(numero>=0){
        numero--;
        resultado *= numero;
    }

    printf("El factorial de %i es %li.\n", numero, resultado);

    return 0;
}
```

start

¿Dónde está la versión corregida?

```
act3.c
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18

int numero;
printf("Ingrese un número:\n");
scanf("%i", &numero);
long int resultado = 1;
while(numero>=0){
    numero--;
    resultado *= numero;
}
printf("El factorial de %i es %li.\n", numero, resultado);
return 0;
```

native process 2748 In: main
(gdb) start
Punto de interrupción temporal 1 at 0x555555554772: file act3.c, line 4.
Starting program: /home/evangelina/Descargas/act3
Temporary breakpoint 1, main () at act3.c:4
(gdb) _

```
Peppermint Terminal

evangelina@evangelina-VirtualBox ~/Descargas $ ls
act1    ejemplo1.c      operadores          tarea2
act1.c   main            operadores.c       tarea3
act2    main4           operadoreslogicos  tiposdevariables-i
act2.c   main.c          operadoreslogicos.c tiposdevariables-i.c
act3.c   mostraryleer    prueba1
ejemplo1  mostraryleer.c tareas1
evangelina@evangelina-VirtualBox ~/Descargas $ gcc -g act3.c -o act3
evangelina@evangelina-VirtualBox ~/Descargas $ gdb ./act3
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 ./act3...hecho.
(gdb) run
```

[Práctica 10 - Google] Peppermint Terminal lat 18:35

gdb

Run

CONCLUSIÓN

- Aprendí usar la terminal gdb dentro de la terminal de Linux. Con el comando `gcc -g _____ -o _____`, para compilar el programa y luego usando. Finalmente para entrar a correrlo utilice `gdb ./_____`.
- Para lograr manejar la terminal de gdb utilice los comandos: `run` , `list (l)`, `quit (q)`, `CTRL+X+a`, `start`, `next (n)`, `CTRL+l`, `break (b)`, `print (p)`, `display`.
- Utilizaremos el comando `gcc -w _____ -O _____ -lm`
- Aprendí que dentro de la terminal gbd nos ayuda a saber dentro de la terminal de Linux donde tenemos errores en el código.