

Laboratorios de computación salas A y B

Marco Antonio Martinez Quintana

Profesor:

Fundamentos de Programación

Asignatura:

3

Grupo:

10

No de Práctica(s):

Enrique Ichazo Bautista

Integrante(s):

*No. de Equipo de
cómputo empleado:*

No aplica

27

No. de Lista o Brigada:

2021-1

Semestre:

4/01/2021

Fecha de entrega:

Observaciones:

CALIFICACIÓN: _____

DEPURACIÓN DE PROGRAMAS

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.

Actividades:

Revisar, a través de un depurador, los valores que va tomando una variable en un programa escrito en C, al momento de ejecutarse.

Utilizando un depurador, revisar el flujo de instrucciones que se están ejecutando en un programa en C, cuando el flujo depende de los datos de entrada.

Introducción

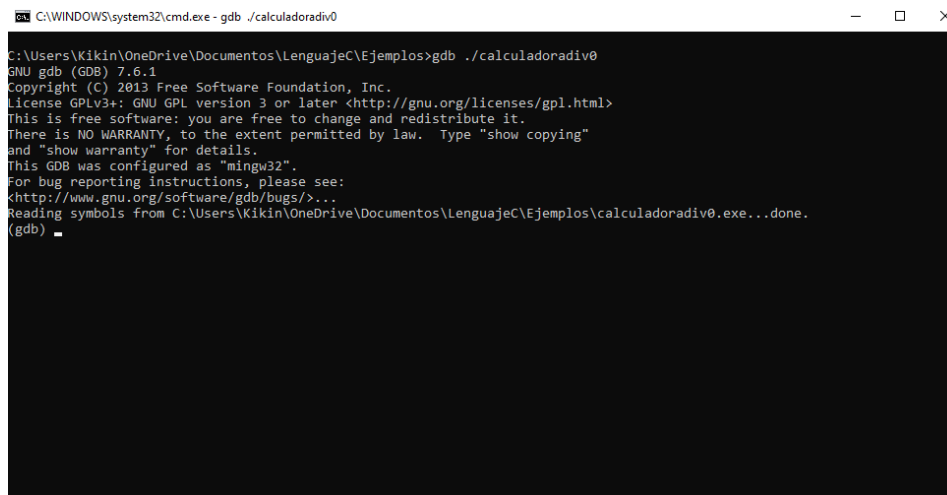
Depurar un programa significa someterlo a un ambiente de ejecución controlado por medio de herramientas dedicadas a ello. Este ambiente permite conocer exactamente el flujo de ejecución del programa, el valor que las variables adquieren, la pila de llamadas a funciones, entre otros aspectos. Es importante poder compilar el programa sin errores antes de depurarlo.

Antes de continuar, es necesario conocer las siguientes definiciones (extraídas del Glosario IEEE610) ya que son parte latente del proceso de Desarrollo de Software:

Error. Se refiere a una acción humana que produce o genera un resultado incorrecto.

Defecto (Fault). Es la manifestación de un error en el software. Un defecto es encontrado porque causa una Falla (failure).

Falla (failure). Es una desviación del servicio o resultado esperado.



```
C:\WINDOWS\system32\cmd.exe - gdb ./calculadoradiv0
GNU gdb (GDB) 7.6.1
Copyright (C) 2013 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 "mingw32".
For bug reporting instructions, please see:
<http://www.gnu.org/software/gdb/bugs/>...
Reading symbols from C:\Users\Kikin\OneDrive\Documentos\LenguajeC\Ejemplos\calculadoradiv0.exe...done.
(gdb) _
```

```
C:\WINDOWS\system32\cmd.exe - gdb ./calculadoradiv0

C:\Users\Kikin\OneDrive\Documentos\LenguajeC\Ejemplos>gdb ./calculadoradiv0
GNU gdb (GDB) 7.6.1
Copyright (C) 2013 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 "mingw32".
For bug reporting instructions, please see:
<http://www.gnu.org/software/gdb/bugs/>...
Reading symbols from C:\Users\Kikin\OneDrive\Documentos\LenguajeC\Ejemplos\calculadoradiv0.exe...done.
(gdb) list
1      #include<stdio.h>
2      int main()
3      {
4          printf("\t\tCalculadora\n\n\t\t");
5
6          //Variables
7          int n1,n2,sum,rest,mul,div,mod;
8          char au=163;
9          char ao=162;
10
(gdb) _
```

```
C:\WINDOWS\system32\cmd.exe - gdb ./calculadoradiv0

1      #include<stdio.h>
2      int main()
3      {
4          printf("\t\tCalculadora\n\n\t\t");
5
6          //Variables
7          int n1,n2,sum,rest,mul,div,mod;
8          char au=163;
9          char ao=162;
10
(gdb) b
No default breakpoint address now.
(gdb) delete
(gdb) b5
Undefined command: "b5". Try "help".
(gdb) b6
Undefined command: "b6". Try "help".
(gdb) cls
Undefined command: "cls". Try "help".
(gdb) b
No default breakpoint address now.
(gdb) b 7
Breakpoint 1 at 0x40142b: file calculadoradiv0.c, line 7.
(gdb) b 8
Note: breakpoint 1 also set at pc 0x40142b.
Breakpoint 2 at 0x40142b: file calculadoradiv0.c, line 8.
(gdb) d 2
(gdb) d 5
No breakpoint number 5.
(gdb) _
```

```
C:\WINDOWS\system32\cmd.exe - gdb ./calculadoradiv0

9          char ao=162;
10
(gdb) b
No default breakpoint address now.
(gdb) delete
(gdb) b5
Undefined command: "b5". Try "help".
(gdb) b6
Undefined command: "b6". Try "help".
(gdb) cls
Undefined command: "cls". Try "help".
(gdb) b
No default breakpoint address now.
(gdb) b 7
Breakpoint 1 at 0x40142b: file calculadoradiv0.c, line 7.
(gdb) b 8
Note: breakpoint 1 also set at pc 0x40142b.
Breakpoint 2 at 0x40142b: file calculadoradiv0.c, line 8.
(gdb) d 2
(gdb) d 5
No breakpoint number 5.
(gdb) info linr 4
Undefined info command: "linr 4". Try "help info".
(gdb) info line 5
Line 5 of "calculadoradiv0.c" is at address 0x40142b <main+27> but contains no code.
(gdb) info line 3
Line 3 of "calculadoradiv0.c" starts at address 0x401410 <main> and ends at 0x40141a <main+10>.
(gdb) info line 7
Line 7 of "calculadoradiv0.c" is at address 0x40142b <main+27> but contains no code.
(gdb) _
```

```
C:\WINDOWS\system32\cmd.exe - gdb ./calculadoradiv0
(gdb) info linr 4
Undefined info command: "linr 4". Try "help info".
(gdb) info line 5
line 5 of "calculadoradiv0.c" is at address 0x40142b <main+27> but contains no code.
(gdb) info line 3
line 3 of "calculadoradiv0.c" starts at address 0x401410 <main> and ends at 0x40141a <main+10>.
(gdb) info line 2
line 7 of "calculadoradiv0.c" is at address 0x40142b <main+27> but contains no code.
(gdb) run
Starting program: C:\Users\Kikin\OneDrive\Documentos\LenguajeC\Ejemplos\./calculadoradiv0.exe
[New Thread 10788.0x2104]
[New Thread 10788.0xa38]
Calculadora

Breakpoint 1, main () at calculadoradiv0.c:8
8      char au=163;
(gdb)
```

```
C:\WINDOWS\system32\cmd.exe - gdb ./calculadoradiv0
(gdb) run
Starting program: C:\Users\Kikin\OneDrive\Documentos\LenguajeC\Ejemplos\./calculadoradiv0.exe
[New Thread 10788.0x2104]
[New Thread 10788.0xa38]
Calculadora

Breakpoint 1, main () at calculadoradiv0.c:8
8      char au=163;
(gdb) c:
A syntax error in expression, near `:'.
(gdb) c
Continuing.
Inserta 2 números separados por espacio:
2 3
La suma de 2 y 3 es: 5
La resta de 2 y 3 es: -1
La multiplicación de 2 y 3 es: 6
La división de 2 y 3 es: 0
El módulo de 2 y 3 es: 2
[Inferior 1 (process 10788) exited normally]
(gdb)
```

```
[Inferior 1 (process 10788) exited normally]
(gdb) s
The program is not being run.
(gdb) n
The program is not being run.
(gdb) _
```

```
C:\WINDOWS\system32\cmd.exe - gdb ./calculadoradiv0
(gdb) run
Starting program: C:\Users\Kikin\OneDrive\Documentos\LenguajeC\Ejemplos\./calculadoradiv0.exe
[New Thread 10788.0x2104]
[New Thread 10788.0xa38]
Calculadora

Breakpoint 1, main () at calculadoradiv0.c:8
8      char au=163;
(gdb) c:
A syntax error in expression, near `:'.
(gdb) c
Continuing.
Inserta 2 números separados por espacio:
2 3
La suma de 2 y 3 es: 5
La resta de 2 y 3 es: -1
La multiplicación de 2 y 3 es: 6
La división de 2 y 3 es: 0
El módulo de 2 y 3 es: 2
[Inferior 1 (process 10788) exited normally]
(gdb) s
The program is not being run.
(gdb) n
The program is not being run.
(gdb) _
```

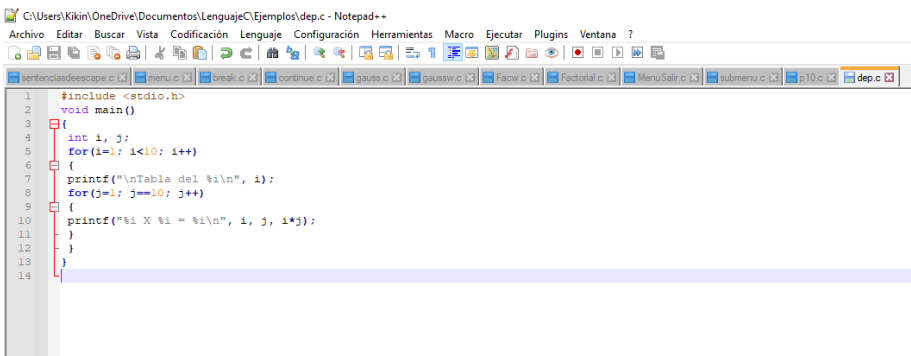
Ejercicios propuestos

Para el siguiente código fuente, utilizar algún entorno de depuración para encontrar la utilidad del programa y la funcionalidad de los principales comandos de depuración, como puntos de ruptura, ejecución de siguiente línea o instrucción.

```
1  #include <stdio.h>
2  void main()
3  {
4      int N, CONT, AS;
5      AS=0;
6      CONT=1;
7      printf("TECLEA UN NUMERO: ");
8      scanf("%i",&N);
9      while(CONT<=N)
10     {
11         AS=(AS+CONT);
12         CONT=(CONT+2);
13     }
14     printf("\nEL RESULTADO ES %i\n", AS);
15 }
16
```

Esté código nos va a servir como contador que surge a partir del 2 en este se sustituirán valores, N va a ser el numero que nosotros indiquemos y AS será la suma de la variable y de la constante CONT cuyo número no sea igual a 1 se le sumará 2 más. Dando así un valor almacenado.

El siguiente programa debe mostrar las tablas de multiplicar desde la del 1 hasta la del 10. En un principio no se mostraba la tabla del 10, luego después de intentar corregirse sin un depurador dejaron de mostrarse el resto de las tablas. Usar un depurador de C para averiguar el funcionamiento del programa y corregir ambos problemas.



```
1  #include <stdio.h>
2  void main()
3  {
4      int i, j;
5      for(i=1; i<10; i++)
6      {
7          printf("\nTabla del %i\n", i);
8          for(j=1; j<10; j++)
9          {
10             printf("%i X %i = %i\n", i, j, i*j);
11         }
12     }
13 }
14
```

```
C:\WINDOWS\system32\cmd.exe

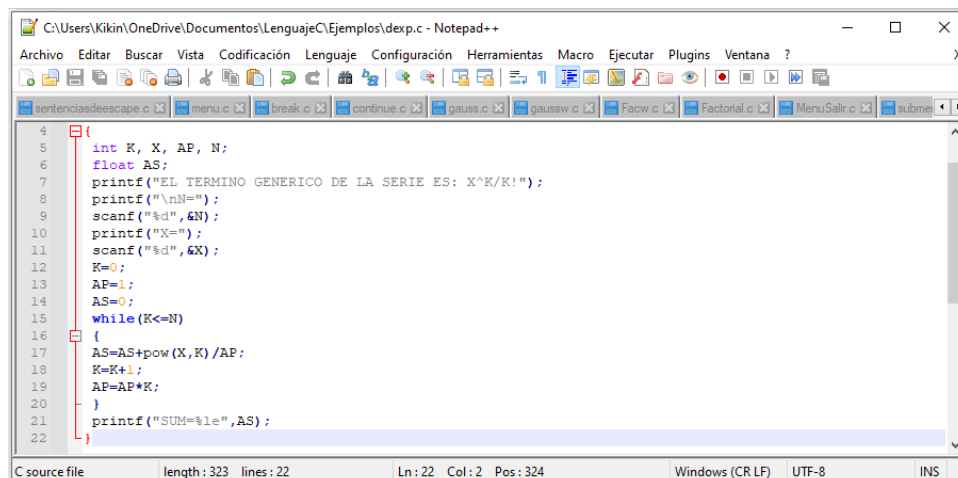
C:\Users\Kikin\OneDrive\Documentos\LenguajeC\Ejemplos>gcc dep.c -o dep.exe

C:\Users\Kikin\OneDrive\Documentos\LenguajeC\Ejemplos>dep.exe

Tabla del 1
Tabla del 2
Tabla del 3
Tabla del 4
Tabla del 5
Tabla del 6
Tabla del 7
Tabla del 8
Tabla del 9

C:\Users\Kikin\OneDrive\Documentos\LenguajeC\Ejemplos>_
```

El siguiente programa muestra una violación de segmento durante su ejecución y se interrumpe; usar un depurador para detectar y corregir la falla.



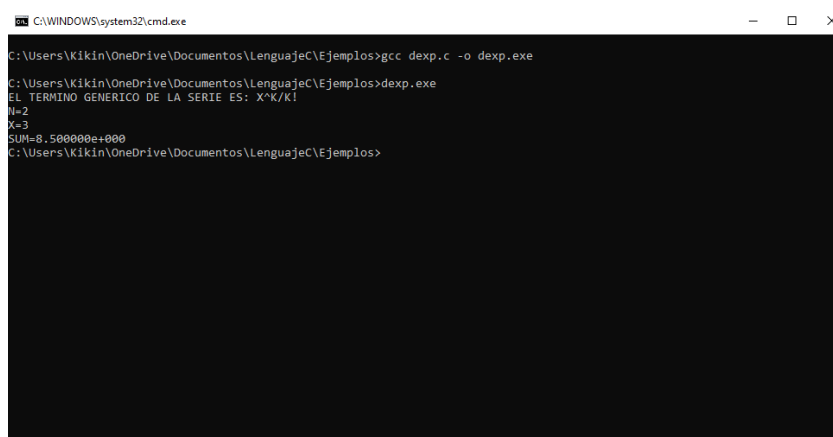
```
C:\Users\Kikin\OneDrive\Documentos\LenguajeC\Ejemplos\dexp.c - Notepad++

Archivo  Editar  Buscar  Vista  Codificación  Lenguaje  Configuración  Herramientas  Macro  Ejecutar  Plugins  Ventana  ?

sentenciasdesescape.c  menu.c  break.c  continue.c  gauss.c  gaussw.c  facw.c  factorial.c  MenuSalir.c  subme

4  {
5  int K, X, AP, N;
6  float AS;
7  printf("EL TERMINO GENERICO DE LA SERIE ES: X^K/K!");
8  printf("\nN=");
9  scanf("%d", &N);
10 printf("X=");
11 scanf("%d", &X);
12 K=0;
13 AP=1;
14 AS=0;
15 while(K<=N)
16 {
17 AS=AS+pow(X,K)/AP;
18 K=K+1;
19 AP=AP*K;
20 }
21 printf("SUM=%le", AS);
22 }
```

(ya corregido)



```
C:\WINDOWS\system32\cmd.exe

C:\Users\Kikin\OneDrive\Documentos\LenguajeC\Ejemplos>gcc dexp.c -o dexp.exe

C:\Users\Kikin\OneDrive\Documentos\LenguajeC\Ejemplos>dexp.exe
EL TERMINO GENERICO DE LA SERIE ES: X^K/K!
N=2
K=3
SUM=8.500000e+000
C:\Users\Kikin\OneDrive\Documentos\LenguajeC\Ejemplos>
```

Conclusión:

Tras acabar la práctica es evidente que un depurador es una de las herramientas mas útiles para un programador, ya que es fundamental poder encontrar y resolver errores de una forma práctica y esto es justo lo que te ofrece un depurador, por lo que aprender a utilizarlo de forma correcta es extremadamente importante ya que estoy seguro de que en la vida profesional será algo que usaré constantemente, por lo que concluyó que esta práctica es de las más importantes de todo el curso, porque nos esta dando un arma demasiado importante para defendernos en un futuro.

Bibliografía

El lenguaje de programación C. Brian W. Kernighan, Dennis M. Ritchie, segunda edición, USA, Pearson Educación 1991.