



Carátula para entrega de prácticas

Facultad de Ingeniería

Laboratorio de docencia

Laboratorios de computación salas A y B

Marco Martínez Quintana

Profesor:

Fundamentos de la programación

Asignatura:

36

Grupo:

10

No de Práctica(s):

Jimena Hernández García

Integrante(s):

*No. de Equipo de
cómputo empleado:*

No aplica

23

No. de Lista o Brigada:

2021-1

Semestre:

Fecha de entrega:

Observaciones:

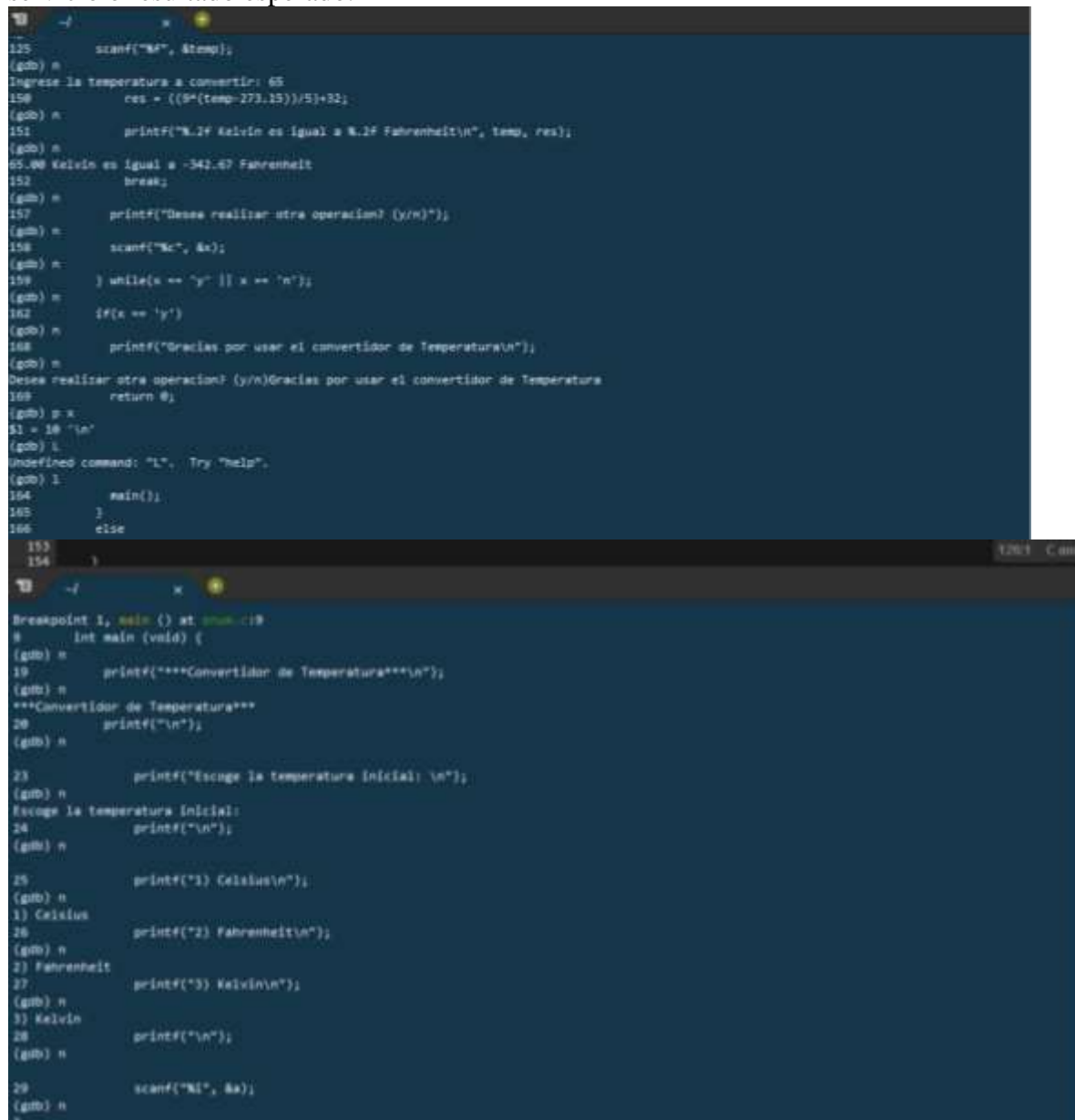
CALIFICACIÓN: _____

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.



```
125 scanf("%d", &temp);
(gdb) n
Ingrese la temperatura a convertir: 65
150 res = ((5*(temp-273.15))/5)+32;
(gdb) n
151 printf("8.24 Kelvin es igual a 8.24 Fahrenheit\n", temp, res);
(gdb) n
85.00 Kelvin es igual a 342.67 Fahrenheit
152 break;
(gdb) n
157 printf("Desea realizar otra operacion? (y/n)");
(gdb) n
158 scanf("%c", &x);
(gdb) n
159 } while(x == 'y' || x == 'n');
(gdb) n
162 if(x == 'y')
(gdb) n
168 printf("Gracias por usar el convertidor de temperatura\n");
(gdb) n
Desea realizar otra operacion? (y/n)Gracias por usar el convertidor de temperatura
169 return 0;
(gdb) p x
x = 10 '\n'
(gdb) l
undefined command: "l". Try "help".
(gdb) l
164 main();
165 }
166 else

153
154 }
```

```
Breakpoint 1, main () at main.c:9
9 int main (void) {
(gdb) n
19 printf("***Convertidor de Temperatura***\n");
(gdb) n
***Convertidor de Temperatura***
20 printf("\n");
(gdb) n

23 printf("Escriba la temperatura inicial: \n");
(gdb) n
Escriba la temperatura inicial:
24 printf("\n");
(gdb) n

25 printf("1) Celsius\n");
(gdb) n
1) Celsius
26 printf("2) Fahrenheit\n");
(gdb) n
2) Fahrenheit
27 printf("3) Kelvin\n");
(gdb) n
3) Kelvin
28 printf("\n");
(gdb) n

29 scanf("%d", &a);
(gdb) n
3
```

```

125     scanf("%d", &temp);
(gdb) n
Ingresar la temperatura a convertir: 45
126     res = ((9*(temp-32))/5)+32;
(gdb) n
127     printf("4.5° Kelvin es igual a 4.2° Fahrenheit\n", temp, res);
(gdb) n
45.00 Kelvin es igual a -342.67 Fahrenheit
128     break;
(gdb) n
129     printf("Desee realizar otra operacion? (y/n)");
(gdb) n
130     scanf("%c", &x);
(gdb) n
131     while(x == 'y' || x == 'n');
(gdb) n
132     if(x == 'y')
(gdb) n
133     printf("Gracias por usar el convertidor de Temperatura\n");
(gdb) n
Desee realizar otra operacion? (y/n)Gracias por usar el convertidor de Temperatura
134     return 0;
(gdb) p x
x1 = 10 '\n'
(gdb) l
Undefined command: "l". Try "help".
(gdb) l
135     main();
136     }
137     else
138     {
139         break;
140     case 45: //Convertir K a F
141         res = ((9*(temp-273.15))/5)+32;
142         printf("4.5° Kelvin es igual a 4.2° Fahrenheit\n", temp, res);
143         break;
144     }
145 }
146
147 int main()
148 {
149     printf("Desee realizar otra operacion? (y/n)");
150     scanf("%c", &x);
151     while(x != 'y' || x != 'n');
152
153     if(x == 'y')
154     {
155         main();
156     }
157     else
158     {
159         printf("Gracias por usar el convertidor de Temperatura\n");
160         return 0;
161     }
162 }
172 }

```

Conclusiones: Es muy importante el conocer como se utiliza el comando GDB para poder depurar programas para mantener mayor orden en nuestra ejecución de programas.