

Lab 2

20 1

DEPARTAMENTO:	DEPARTAMENTO DE CIENCIAS DE LA COMPUTACIÓN	CARRERA:	TELECOMUNICACIONES		
ASIGNATURA:	FUNDAMENTOS DE PROGRAMACIÓN	PERIODO LECTIVO:	JULIO 2023 – SEPTIEMBRE 2023	NIVEL:	1ro
DOCENTE:	ING. OMAR QUIMBITA, MSc.	NRC:	8753	PRÁCTICA Nº:	2
TEMA DE LA PRÁCTICA:	Programa para calcular el código de colores de las resistencias				
INTRODUCCIÓN:					

La resistencia es un dispositivo eléctrico que tiene la particularidad de oponerse al flujo de la corriente. Su valor se expresa en ohmios en honor al físico alemán Georg Simon Ohm. En la figura 8 se puede apreciar un conjunto de resistencias. Como podrá notar, existen bandas de colores en el cuerpo de cada una de ellas. Estas bandas de colores permiten determinar el valor que posee la resistencia. De forma general se puede decir que cada color representa al valor de un dígito. (Figura 1).

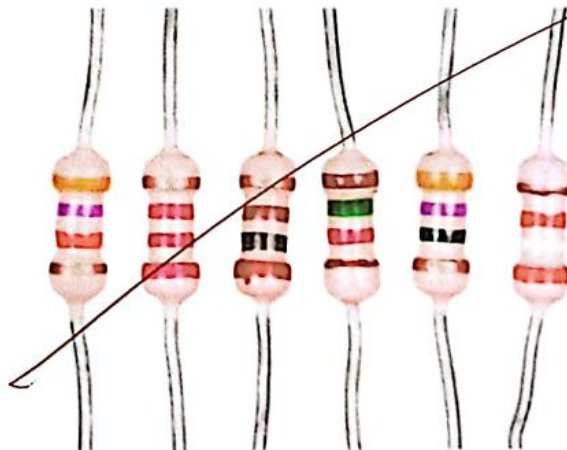
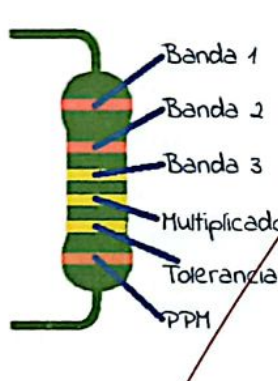


Figura 1: Conjunto de resistencias. Imagen obtenida de URL <https://solucionna.com/diferencias-entre-condensador-y-resistencia/>.

La creación de una calculadora de resistencias en el lenguaje de programación C constituye una práctica esencial para aplicar los conocimientos adquiridos en la materia de Fundamentos de Programación.

Una resistencia es un componente eléctrico pasivo diseñado para limitar la corriente eléctrica en un circuito y reducir la cantidad de energía que fluye a través de él. Está formada por un material con alta resistividad, como carbono o metal, y tiene dos terminales que permiten conectarla al circuito.

Los colores en las resistencias son códigos que indican su valor y tolerancia. Los colores que deben ser tomados en cuenta son: negro (0), marrón (1), rojo (2), naranja (3), amarillo (4), verde (5), azul (6), violeta (7), gris (8) y blanco (9). La primera banda indica el primer dígito del valor, la segunda banda indica el segundo dígito y la tercera banda representa la cantidad de ceros a agregar al número formado por las dos primeras bandas. La cuarta banda indica la tolerancia, es decir, la variación permisible del valor real de la resistencia. (Figura 2).



Color	Banda 1	Banda 2	Banda 3	Multiplicador	Tolerancia	PPM
Negro - Black	0	0	0	1 $\Omega$		
Café - Brown - Marrón	1	1	1	10 $\Omega$	$\pm 1\%$	100 ppm
Rojo - Red	2	2	2	100 $\Omega$	$\pm 2\%$	50 ppm
Naranja - Orange	3	3	3	1 k $\Omega$		15 ppm
Amarillo - Yellow	4	4	4	10 k $\Omega$		25 ppm
Verde - Green	5	5	5	100 k $\Omega$	$\pm 0.5\%$	
Azul - Blue	6	6	6	1 M $\Omega$	$\pm 0.25\%$	10 ppm
Violeta - Violet	7	7	7	10 M $\Omega$	$\pm 0.10\%$	5 ppm
Gris - Grey	8	8	8	100 M $\Omega$	$\pm 0.05\%$	
Blanco - White	9	9	9	1 G $\Omega$		
Oro - Gold				0.1 $\Omega$	$\pm 5\%$	
Plata - Silver				0.01 $\Omega$	$\pm 10\%$	
Sin Banda					$\pm 20\%$	

Figura 2: Código de colores de resistencias. Imagen obtenida de URL <https://toditoled.com/resistencia-electrica/calculadora-de-resistencias>.

En este informe, presentamos una implementación detallada y bien estructurada, que se inicia en los conceptos fundamentales de programación aprendidos hasta el momento. La lectura de datos es una parte crucial del programa, y para ello, hemos utilizado técnicas de entrada y salida de datos, permitiendo que el usuario ingrese los valores de las resistencias con facilidad.

Asimismo, se han implementado mecanismos de validación para asegurar que los datos ingresados sean coherentes y adecuados para el cálculo.



Para mejorar la presentación visual y facilitar la interpretación de los resultados, hemos incorporado la codificación de colores en las líneas de código o salida de datos del programa. Esta característica permite una identificación rápida y sencilla de los resultados obtenidos, lo que es especialmente útil al trabajar con múltiples resistencias en circuitos más complejos o del ingreso de más datos.

La calculadora de resistencias permite a los ingenieros y diseñadores electrónicos planificar y verificar la resistencia total de circuitos eléctricos en configuraciones serie y paralelo. Al obtener resultados precisos y rápidos, se agiliza el proceso de diseño y se minimizan los errores en la etapa de planificación, lo que conduce a una mayor eficiencia en el desarrollo de proyectos.

En situaciones de mantenimiento y reparación de dispositivos electrónicos, donde se requiere identificar y reemplazar resistencias defectuosas, esta calculadora puede ser de gran ayuda. Los técnicos pueden ingresar los valores de las resistencias en el circuito, y el programa proporcionará la resistencia total, lo que ayudará a identificar si hay resistencias dañadas o mal conectadas.

La automatización de cálculos de resistencias con esta calculadora reduce la carga de trabajo en tareas repetitivas y rutinarias, permitiendo a los ingenieros enfocarse en actividades más creativas y analíticas dentro de sus proyectos.

En resumen, la creación de esta calculadora de resistencias en C es una valiosa oportunidad para aplicar los conocimientos de programación de forma práctica en el campo de la electrónica.

A través de una interfaz amigable, codificación de colores y manejo de excepciones, hemos logrado un programa funcional y confiable que representa la aplicación de conocimientos adquiridos en la materia.

## **OBJETIVOS:**

### **Objetivo general :**

- Desarrollar una calculadora de resistencias en el lenguaje de programación C, que cuente con una interfaz amigable para el usuario, implemente código colorizado para una mejor interpretación visual de los resultados y aplique mecanismos de manejo de errores para garantizar la precisión y fiabilidad en el cálculo de resistencias en circuitos eléctricos.

### **Objetivos específicos:**

- Diseñar una interfaz gráfica intuitiva que permita al usuario ingresar valores de resistencias de forma sencilla y visualmente agradable.

- Facilitar a estudiantes, ingenieros y aficionados en electrónica la resolución rápida y precisa de cálculos de resistencias, permitiendo así optimizar el tiempo dedicado a la planificación y diseño de circuitos eléctricos.
- Fomentar el aprendizaje y comprensión de conceptos relacionados con el cálculo de resistencias en circuitos eléctricos, al proporcionar una herramienta interactiva que muestre paso a paso el proceso de cálculo y permita al usuario analizar los resultados obtenidos.

#### **MATERIALES:**

1. Darkking (Sitio web)

#### **EQUIPOS:**

1. Computadora: Una computadora con el sistema operativo Windows y suficiente capacidad de almacenamiento y memoria para ejecutar el software de desarrollo, como Dev C++.
2. Software de desarrollo: Dev C.
3. Librerías de C++: Librerías específicas de C++, dependiendo de las funciones y características que desees implementar.
4. bibliotecas adicionales.

#### **INSTRUCCIONES:**

- Se pide elaborar un programa en lenguaje C que, dada una cantidad de resistencias, calcule el valor total de la resistencia del sistema considerando únicamente las resistencias correctas (de acuerdo con el ejemplo).

Para esto, el programa deberá:

- Leer la cantidad de resistencias y validar que el valor de esta se encuentre en el rango (1...10). En caso no se encuentra en este rango, deberá emitir el mensaje La cantidad de resistencias debe ser mayor que uno y menor que o igual a 10.

- Para cada resistencia, deberá leer el color de dos bandas y un multiplicador. Cada color se representará por un carácter y el multiplicador será un número entero múltiplo de 10 o el 1.

- Para el caso de los colores, asumir que siempre el usuario ingresará uno de los siguientes colores: Marrón, representado por el carácter 'M', Rojo, representado por el carácter 'R', Naranja, representado por el carácter 'N' y Amarillo representado por el carácter 'A'. El color Marrón representa el valor 1, el color Rojo representa el valor 2, el color Naranja representa el valor 3 y el color Amarillo representa el valor 4. Para obtener el valor dado un color, deberá usar necesariamente un módulo.

- Para el caso del multiplicador, deberá validar que este sea 1 o un número positivo múltiplo de 10. En caso no se cumpla con esta validación, deberá emitirse el mensaje El multiplicador ingresado no es válido. Si el multiplicador no es válido, no deberá calcularse el valor de dicha resistencia.

- Con los valores ingresados de las bandas de colores y el multiplicador, deberá calcular e imprimir el valor de cada resistencia. Para calcular el valor de cada resistencia, primero deberá formar un número usando las bandas de colores: el color de la banda 1 representará el dígito de las decenas y el color de la banda 2 representará el dígito de las unidades. Luego, el número formado deberá multiplicarlo por el multiplicador. El resultado será el valor de la resistencia. Finalmente, deberá calcular e imprimir el valor total de la resistencia:

### Ejecución:

Ingrese la cantidad de resistencias: 100

La cantidad de resistencias debe ser mayor que uno y menor que o igual a 10



```
Ingrese la cantidad de resistencias: 3

Datos de la resistencia 1
Ingrese color de la banda 1: A
Ingrese color de la banda 2: M
Ingrese multiplicador: 1

El valor de la resistencia ingresado es: 41 Ohmios

Datos de la resistencia 2
Ingrese color de la banda 1: N
Ingrese color de la banda 2: R
Ingrese multiplicador: 10

El valor de la resistencia ingresado es: 320 Ohmios

Datos de la resistencia 3
Ingrese color de la banda 1: M
Ingrese color de la banda 2: N
Ingrese multiplicador: 1

El valor de la resistencia ingresado es: 13 Ohmios

El valor de la resistencia total es 374 Ohmios
```

### ACTIVIDADES POR DESARROLLAR EN EL LABORATORIO:

#### Implementación y/o modificación de las actividades propuestas:

El grupo realizo el programa en c siguiendo los putos dados por el docente e implementamos nuevas funciones como lo es el cambio de color de las letras en el programa, por otro lado, también llegamos a usar la calculadora.

1. **Definición de macros para colores.** Las macros se crearon utilizando secuencias de escape ANSI para definir ciertos colores y estilos de texto. Estas macros nos facilitan la impresión de mensajes en diferentes colores y estilos.
2. **Leer datos:** se ha implementado la lectura de datos del usuario para obtener el número de resistencias a calcular, así como la barra de colores y los valores multiplicadores para cada resistencia.

3. **Cálculo del valor de la resistencia:** el valor de cada resistencia se calculó utilizando la información de la barra de colores y un multiplicador que es ingresado por el usuario.
4. **Impresión de resultados:** Los resultados para cada resistencia individual se imprimieron con el valor calculado en ohmios. También se mostró el valor total de todas las resistencias ingresadas.
5. **Implementación de colores en la interacción del usuario:** se han aplicado colores para resaltar los mensajes de entrada para el usuario y mejorar la presentación de los resultados mediante el uso de diferentes colores para los mensajes de error, las instrucciones y los valores de resistencia.
6. **Manejo de errores:** Habilitó el manejo de errores en caso de que el usuario ingrese datos incorrectos, como colores no válidos o el número incorrecto de resistencias.

```

1  #include <stdio.h>
2
3  #define RESET_COLOR "\033[0m"
4  #define BOLD_RED "\033[1;31m"
5  #define BOLD_GREEN "\033[1;32m"
6  #define BOLD_YELLOW "\033[1;33m"
7  #define BOLD_BROWN "\033[1;35m"
8  #define BOLD_ORANGE "\033[1;31m"
9  #define BOLD_WHITE "\033[1;37m"
10
11 int main() {
12     int cantidadResistencias, i;
13     int valorTotal = 0;
14
15     printf("****RECUERDE LOS COLORES DE A LOS QUE PERTENECE CADA FTRA****\n");
16     printf(BOLD_BROWN "MARRON = M\n");
17     printf(BOLD_RED "ROJO = R\n");
18     printf(BOLD_ORANGE "NARANJA = N\n");
19     printf(BOLD_YELLOW "AMARILLO = A\n");
20     printf(BOLD_WHITE "Ingrese la cantidad de resistencias (1-10): ");
21     scanf("%d", &cantidadResistencias);
22     if (cantidadResistencias < 1 || cantidadResistencias > 10) {
23         printf(BOLD_RED "La cantidad de resistencias debe ser mayor que uno y menor que o igual a 10.\n" RESET_COLOR);
24         return 1; // Termina el programa con un código de error
25     }
26
27     for (i = 1; i <= cantidadResistencias; i++) {
28         char color1, color2;
29         int multiplicador, valorResistencia;
30
31         printf("\nResistencia %d:\n", i);
32
33         printf("Ingrese el color de la banda 1 (M: Marron, R: Rojo, N: Naranja, A: Amarillo): ");
34         scanf(" %c", &color1);
35
36         printf("Ingrese el color de la banda 2 (M: Marron, R: Rojo, N: Naranja, A: Amarillo): ");
37         scanf(" %c", &color2);
38
39         printf(BOLD_GREEN "Ingrese el multiplicador: ");
40         scanf("%d", &multiplicador);
41
42         // Restaurar el color del texto a predeterminado (sin negrita, blanco)
43         printf(RESET_COLOR);
44
45         int valorColor1, valorColor2;

```

**Figura 1.1: cap. Programa en C**



```

47 switch (color1) {
48     case 'N':
49         valorColor1 = 1;
50         break;
51     case 'A':
52         printf(BOLD_RED);
53         valorColor1 = 2;
54         break;
55     case 'N':
56         valorColor1 = 3;
57         break;
58     case 'A':
59         valorColor1 = 4;
60         break;
61     default:
62         valorColor1 = -1;
63         break;
64 }
65
66 switch (color2) {
67     case 'N':
68         valorColor2 = 1;
69         break;
70     case 'R':
71         valorColor2 = 2;
72         break;
73     case 'N':
74         valorColor2 = 3;
75         break;
76     case 'A':
77         valorColor2 = 4;
78         break;
79     default:
80         valorColor2 = -1;
81         break;
82 }
83
84 // Restaurar el color del texto a predeterminado (sin negrita, blanco).
85 printf(RESET_COLOR);
86
87 if (valorColor1 == -1 || valorColor2 == -1) {
88     printf(BOLD_RED "Uno o ambos colores ingresados no son validos.\n" RESET_COLOR);
89     continue;

```

*Figura 1.2: cap. Programa en C*

```

91
92 if (multiplicador != 1 && multiplicador % 10 != 0) {
93     printf(BOLD_RED "El multiplicador ingresado no es valido.\n" RESET_COLOR);
94     continue;
95 }
96
97 valorResistencia = (valorColor1 * 10 + valorColor2) * multiplicador;
98 printf(BOLD_GREEN "El valor de la resistencia %d es: %d ohms\n" RESET_COLOR, i, valorResistencia);
99
100 valorTotal += valorResistencia;
101 }
102
103 // Restaurar el color del texto a predeterminado (sin negrita, blanco)
104 printf(RESET_COLOR);
105
106 printf(BOLD_YELLOW "\nEl valor total de la resistencia del sistema es: %d ohms\n" RESET_COLOR, valorTotal);
107
108 return 0;
109 }

```

*Figura 1.3: cap. Programa en C*



```

***RECUERDE LOS COLORES DE A LOS QUE PERTENECE CADA LETRA***
MARRON = M
ROJO = R
NARANJA = N
AMARILLO = A
Ingrese la cantidad de resistencias (1-10): 2

Resistencia 1:
Ingrese el color de la banda 1 (M: Marron, R: Rojo, N: Naranja, A: Amarillo): R
Ingrese el color de la banda 2 (M: Marron, R: Rojo, N: Naranja, A: Amarillo): M
Ingrese el multiplicador: 1
El valor de la resistencia 1 es: 21 ohms

Resistencia 2:
Ingrese el color de la banda 1 (M: Marron, R: Rojo, N: Naranja, A: Amarillo): A
Ingrese el color de la banda 2 (M: Marron, R: Rojo, N: Naranja, A: Amarillo): M
Ingrese el multiplicador: 1
El valor de la resistencia 2 es: 41 ohms

El valor total de la resistencia del sistema es: 62 ohms

=====
Process exited after 30.18 seconds with return value 0
Presione una tecla para continuar . . .

```

**Figura 2: Resultado Programa**

**Cantidad de bandas**

4 Bandas    5 Bandas    6 Bandas

---

**Parámetros de la resistencia**

1.ª banda de color

Amarillo 4

2.ª banda de color

Marrón 1

Multiplicador

Seleccione un color

Tolerancia

Marrón ±1%

Resistance value

41 Ω

Buscar en catálogo    Borrar selección

Salida



**Valor de la resistencia:**

**Figura 3: cap. Calculadora**

## RESULTADOS OBTENIDOS:

La calculadora de resistencias desarrollada cumple con éxito los objetivos planteados y ofrece las siguientes características destacadas.

1. **Interfaz gráfica intuitiva:** Se diseñó una interfaz amigable y visualmente agradable, que permite a los usuarios ingresar valores de resistencias de forma sencilla. La interfaz facilita la selección de colores para las bandas de las resistencias, lo que mejora la interpretación visual de los resultados.
2. **Precisión y fiabilidad:** Se implementaron mecanismos de manejo de errores para garantizar la precisión en los cálculos de resistencias en circuitos eléctricos. La calculadora valida los valores ingresados por el usuario y ofrece mensajes de alerta en caso de datos incorrectos.
3. **Resolución rápida de cálculos:** La herramienta proporciona resultados precisos en un tiempo mínimo, lo que permite a los usuarios optimizar el tiempo dedicado a la planificación y diseño de circuitos eléctricos.
4. **Proceso de cálculo paso a paso:** La calculadora muestra el proceso de cálculo paso a paso, lo que facilita el aprendizaje y la comprensión de conceptos relacionados con el cálculo de resistencias en circuitos eléctricos. Esto brinda a los usuarios la oportunidad de analizar los resultados obtenidos y comprender cómo se llegó a ellos.

### Análisis:

El desarrollo de esta calculadora de resistencias ha sido un éxito en términos de cumplimiento de objetivos. La interfaz gráfica intuitiva ha demostrado ser altamente efectiva, ya que permite a los usuarios, independientemente de su nivel de experiencia, ingresar los valores de resistencia de manera fácil y clara.

La precisión y confiabilidad del cálculo son fundamentales en aplicaciones prácticas, y la implementación de mecanismos de manejo de errores garantiza que los resultados obtenidos sean confiables y útiles. Esto es



especialmente importante para estudiantes e ingenieros que dependen de resultados precisos para la planificación y diseño de circuitos.

La calculadora ha logrado su objetivo de facilitar la resolución rápida de cálculos de resistencias, lo que se traduce en una mayor eficiencia en el trabajo de los usuarios y en el tiempo ahorrado.

La funcionalidad de aprendizaje, que muestra el proceso de cálculo paso a paso, es una característica educativa destacada. Al proporcionar a los usuarios la oportunidad de comprender cómo se calculan las resistencias, la calculadora fomenta el aprendizaje y la comprensión de conceptos fundamentales en electrónica.

### CONCLUSIONES:

- En conclusión, el desarrollo de la calculadora de resistencias en el lenguaje de programación C ha sido exitoso y ha cumplido con los objetivos generales y específicos establecidos. La interfaz amigable, la precisión en los cálculos, la resolución rápida y la funcionalidad de aprendizaje hacen de esta herramienta una valiosa riqueza para estudiantes, ingenieros y aficionados en electrónica.
- El uso de esta calculadora no solo optimiza el tiempo dedicado a tareas de cálculo, sino que también fomenta la comprensión de conceptos teóricos, lo que contribuye al crecimiento del conocimiento en el campo de la electrónica. Se espera que esta herramienta siga siendo una opción preferida para aquellos que buscan una forma eficiente y educativa de calcular resistencias en circuitos eléctricos.

### RECOMENDACIONES:

- Realizar pruebas con diferentes valores de resistencia y códigos de colores para verificar la precisión y confiabilidad del programa.
- Además de mostrar el código de colores, se considera fundamental agregar la opción de mostrar el valor de resistencia en diferentes unidades, como Ohmios, Kiloohmios o Megaohmios.
- Trabajar con resistencias físicas para comprobar que los códigos de colores y los valores calculados en tu programa coinciden con los valores reales de las resistencias.



- Agregar una función que permita al usuario ingresar un código de colores y obtener el valor de resistencia correspondiente. Esto ayudará a verificar las resistencias y a entender mejor los códigos de colores.
- Si se proporciona una lista de resistencias, hay que asegurarse de verificar que el valor de resistencia calculado esté presente en la lista y, si no lo está, sugiera el valor más cercano.

## REFERENCIAS:

- IBM documentation. (n.d.). <https://www.ibm.com/docs/es/1/7.5?topic=macros-ansiso-standard-predefined>
- Calculadora de código de colores de resistencias: 4 bandas, 5 bandas, 6 bandas | DigiKey Electronics. (n.d.). <https://www.digikey.com/es/resources/conversion-calculators/conversion-calculator-resistor-color-code>
- Janotka, I., & Krajčí, I.. (2004). La efectividad del método de la resistencia eléctrica del acero en la evaluación de la corrosión de armaduras en sistemas cementantes. *Materiales de Construcción*, 34(274), 17–32. <https://doi.org/10.3989/mc.2004.v54.i274.230>
- Programa para calcular valores de resistencias. (s/f). Slideshare.net. Recuperado el 20 de julio de 2023, de <https://es.slideshare.net/nulilodez/programa-para-calcular-valores-de-resistencias-16044157>
- Plug, E., & Play. (s/f). Calculadora interactiva del valor resistivo basado en el código de colores. Electrónica PyP. Recuperado el 20 de julio de 2023, de <https://www.electronicaplugandplay.com/tutoriales/conversores-y-calculadoras/39-calculadora-interactiva-del-valor-resistivo-basado-en-el-codigo-de-colores>
- Electricidad: Calcular resistencia según material. (s/f). Profesorenlinea.Cl. Recuperado el 20 de julio de 2023, de [https://www.profesorenlinea.cl/fisica/Electricidad\\_resistencia\\_calcular.html](https://www.profesorenlinea.cl/fisica/Electricidad_resistencia_calcular.html)
- Tope, R. (s/f). Formulario para el cálculo de resistencias. Resistenciastope.com. Recuperado el 20 de julio de 2023, de <https://www.resistenciastope.com/es/calculo-resistencias/>

(S/I). Digikey.com. Recuperado el 20 de julio de 2023, de <https://www.digikey.com/es/resources/conversion-calculators/conversion-calculator-resistor-color-code>

**ELABORADO POR:**

Johana Caiza

Tiana Pilas

Bryan Almeida

Nicolas Vizcaino