



Laboratorio Clase 2

- Aplicación del procedimiento de desarrollo  
De software



Recordemos los pasos del proceso:

ANÁLISIS

DISEÑO Y DESARROLLO

CODIFICACION

PRUEBA



## Laboratorio Clase 2

*Problema Ej. Página 86 Bronson.*

*La resistencia eléctrica  $r$ , de un alambre metálico, en ohmios, esta dada por la formula  $r=(m.l)/a$ , donde  $m$  es la resistividad del metal;  $l$  es el largo del metal, en pies; y  $a$  es el área de corte transversal del alambre, en circular-mils. Usando esta info escriba un programa en c++ para calcular la resistencia de un alambre que mide 125 pies de largo, tiene un area de corte transversal de 500 mils circulares y es de cobre. La resistividad del cobre es 10.4*

### **Primer paso: Análisis básico:**

Encontrar palabras como: calcular, imprimir, determinar, encontrar o comparar.

En este caso la salida debe ser: “calcular la resistencia del alambre”  
Por lo tanto es una sola salida.



## Laboratorio Clase 2

Una vez obtenida la salida, se debe analizar las entradas necesarias para poder producir esa salida.

Es importante distinguir entre ELEMENTO DE ENTRADA y VALOR DE ENTRADA, debemos distinguir los elementos necesarios para el calculo de la salida pero no lo valores que tienen ellos.

Los elementos de entrada son la resistividad  $m$ , el largo  $l$  y el area  $a$ .

Podemos ver que el algoritmo para la solución esta dado por la formula  
$$R=(m.l)/a$$

### **Paso 2: Desarrollar una solución**

El algoritmo nos da la formula, ahora debe refinarse paso a paso en detalle como han de combinarse las entradas para producir la salida.



## Laboratorio Clase 2

### Pseudo código:

*Asignar valores a  $m$ ,  $l$  y  $a$ .*

*Calcular la resistencia usando la formula  $r=(m.l)/a$*

*Mostrar la resistencia*

Ahora corresponde hacer una prueba manual en lápiz y papel del algoritmo para verificar que cumpla el objetivo.

Esto nos va a permitir contrastar en la etapa final de prueba del programa si el resultado es el mismo.

### Paso 3: Codificar la solución

Debido al análisis minucioso la codificación es simple. Se necesita **declarar variables para cada entrada y para la salida.**

Se necesita además inicializar variables en valores correctos. Calcular la resistencia, e imprimir el resultado.



## Laboratorio Clase 2

### Codificación:

```
#include <iostream>
using namespace std;
```

```
int main( )
{
```

```
    double resistividad, area, longitud, resistencia;
```

```
    resistividad = 10.4;
```

```
    area = 500;
```

```
    longitud = 125;
```

```
    resistencia = (resistividad * longitud) / area;
```

```
    cout << "La resistencia del alambre es "
         << resistencia << endl;
```

```
    system("pause");
    return 0;
}
```

Instrucción de  
declaración múltiple

Instrucciones de  
asignación

**FALTAN  
COMENTARIOS!!!!!!**

La salida es:

***La resistencia del alambre es 2.6***



## Laboratorio Clase 2

### **Paso 4: Prueba y corrección**

El objetivo de esta etapa es verificar que funciona en forma correcta y en realidad satisface los requerimientos.

Aquí verificamos la respuesta del programa, al calculo realizado en forma manual.

Debido a que este programa solo realiza un calculo, y que la salida producida por la ejecución de prueba coincide con el calculo manual se tiene un buen grado de confianza de que puede ser usado para calcular la resistencia de otros valores de entrada.



Laboratorio Clase 2

**Ejercicio:**

**EJERCICIO 2 - página 89 Bronson.**

**Aparte de lo indicado en el enunciado:**

**COMENTAR BIEN EL PROGRAMA**

**IMPRIMIR EN PANTALLA LA DIRECCION FISICA DE EL/LOS  
RESULTADOS**

**PAUSAR EL PROGRAMA PARA VER RESULTADO system("pause");**

**EJECUTARLO**





Laboratorio Clase 2

2. Suponga que tiene que escribir un programa para calcular el valor de la *distancia*, en millas, dada la relación:

$$\text{distancia} = \text{velocidad} * \text{tiempo transcurrido}$$

- a. Para este problema de programación, ¿cuántas salidas se requieren?
- b. ¿Cuántas entradas tiene este problema?
- c. Determine un algoritmo para convertir los elementos de entrada en elementos de salida.
- d. Pruebe el algoritmo escrito para la parte c usando la siguiente muestra de datos: *velocidad* es 55 millas por hora y *tiempo transcurrido* es 2.5 horas.
- e. ¿Cómo debe modificarse el algoritmo que determinó en la parte c si el tiempo transcurrido se diera en minutos en lugar de horas?



Laboratorio Clase 2

Ejercicio:

EJERCICIO 1 - página 89 Bronson.

Aparte de lo indicado en el enunciado:

COMENTAR BIEN EL PROGRAMA

IMPRIMIR EN PANTALLA LA DIRECCION FISICA DE EL/LOS  
RESULTADOS

PAUSAR EL PROGRAMA PARA VER RESULTADO `system("pause");`

EJECUTARLO



Laboratorio Clase 2

1. Suponga que tiene que escribir un programa en C++ para calcular la resistencia total de un circuito en serie. En dicho circuito la resistencia total es la suma de todos los valores de resistencia individuales. Suponga que el circuito consiste en una cantidad de resistores de 56 ohmios, 33 ohmios y 15 ohmios.
  - a. Para este problema de programación, ¿cuántas salidas se requieren?
  - b. ¿Cuántas entradas tiene este problema?
  - c. Determine un algoritmo para convertir los elementos de entrada en elementos de salida. Suponga que la cantidad de resistores de 56 ohmios es  $m$ , la cantidad de resistores de 33 ohmios es  $n$ , la cantidad de resistores de 15 ohmios es  $p$ .
  - d. Pruebe el algoritmo escrito para la parte c usando la siguiente muestra de datos:  
 $m = 17$ ,  $n = 24$  y  $p = 12$ .