

AEDD-Guía Práctica 1: Zinjal y estructura de un programa en C++

Los ejercicios a desarrollar en esta guía práctica están centrados en ejemplos/ejercicios que permiten:

- Conocer la estructura básica de un programa en C++.
- Utilizar el IDE (entorno de desarrollo) Zinjal para la edición, depuración y ejecución de los programas que introducen al uso de números, cadenas, expresiones, variables, asignaciones y entrada/salida de datos.

Nota: trabajaremos sólo con variables enteras (tipo de dato `int`)

1. Utilizando el Zinjal copia el siguiente programa en el editor. Luego, compilalo y ejecutalo. Observando el comportamiento resultante analizar: ¿Qué realiza el programa? ¿Cómo se estructura un programa en C++?

```
#include <iostream>

using namespace std;

int main (void) {
    int nroLegajo;

    cout << "Me dices tu número de legajo?" << endl;
    cin >> nroLegajo;
    cout << "Ingreso de " << nroLegajo << " al Zinjai!" << endl;

    return 0;
}
```

2. Tu misión, si decides aceptarla, es recibir los 3 números que desactivan la bomba del destino final. Para desactivarla simplemente tienes que mostrar los 3 números recibidos en el orden inverso al que los recibiste separados entre sí por guión medio.

Ejemplo:
Ingrese el código de desactivación de la bomba: 33 24 51
El código de desactivación real es 51-24-33

3. Dadas dos variables numéricas a y b, que el usuario debe ingresar, se pide realizar un algoritmo que **intercambie** los valores de ambas variables y muestre cuánto valen ambas al final.

Ejemplo:
Ingrese valor de a: 130
Ingrese valor de b: 45
--Intercambiamos a y b--
a: 45
b: 130

4. Un docente desea saber qué porcentaje de hombres y qué porcentaje de mujeres hay en un grupo de alumnos.

Ejemplo:
Ingrese cantidad de mujeres: 30
Ingrese cantidad de varones: 20
En la comisión hay un 60% de mujeres y un 40% de varones.

5. El siguiente programa transforma una cantidad dada de segundos en la hora y fecha que representan dichos segundos, contando el tiempo transcurrido desde las 00:00:00 horas del día 1 de enero del 2020 (considerando que todos los meses tienen 30 días para simplificar el cálculo de los meses). Sin embargo, el programa contiene muchos errores que se deben corregir para que pueda realizar lo que se pretende.

A fin de arreglarlo, deberás transcribir el código propuesto utilizando Zinjal e intentar ejecutarlo para que el compilador indique los errores encontrados. Luego, deberás realizar las correcciones oportunas para depurar cada uno de esos errores, hasta que no quede ninguno. Atención porque también puede haber errores de lógica, que no son detectados por el compilador.

Cuando hayas corregido todos los errores, puedes ingresar 28045678 como cantidad de segundos a transformar para comprobar que funciona correctamente. En este caso, lo que verás en la ventana de ejecución interactiva será:

```
Inicio del tiempo: 00:00:00 del 01/01/2020
Segundos transcurridos: 28045678
Ese tiempo corresponde a las 14:27:58 del día 25 de 11 de 2020
```

Nota: Observa el tipo de error que te indica el compilador. En la mayoría de los casos te mostrará exactamente la línea donde está el error. No obstante, en unos pocos casos la línea que te mostrará será en la que él ha detectado un error y no en la que realmente se ha producido. Ten en cuenta también que la "correcta" corrección de algunos errores depende de la acción que se pretende realizar, para lo cual es muy conveniente que entiendas qué va realizando el programa paso a paso.

```
#include <iostream>
```

```
using namespace std;
```

```
int main () {
```

```
    // Este programa cambia el formato del tiempo
    long segundos_trans, minutos_trans, horas_trans, dias_trans
    long segs, mins, hors, mes, dia;
```

```
    cout << "Inicio del tiempo: 00:00:00 del 01/01/2020" << endl; cout <<
    "Segundos transcurridos: "
    cin >> segundos_trans;
```

```
    minutos_trans = segundos_trans / 60;
    segundos_trans % 60 = segs;
    horas_trans = minutos_trans / 0;
```

```
    mins = minutos_trans % 60;
```

```
    // dias_trans = horas_trans / 24;
    "hors" = horas_trans % 24;
    mes = dias_trans / 30;
    dia = dias_trans % 30 + 1;
```

```
    cout << "Ese tiempo corresponde a la hora " << hors << ":" << mins <<
    ":" << segs;
    cout << " del día " << dia << " de " << mes << " de 2020" << endl;
```

```
    return 0;
```

```
}
```