

Laboratorio de Computación II

Docentes

- Angel Simón - Profesor
asimon@docentes.frgp.utn.edu.ar
- Brian Lara - Profesor
blara@docentes.frgp.utn.edu.ar
- Verónica Carbonari - Jefe de trabajos prácticos
veronica.carbonari@alumnos.frgp.utn.edu.ar
- Mariano Varela - Jefe de trabajos prácticos
mvarela@docentes.frgp.utn.edu.ar
- Ariel Tapia - Jefe de trabajos prácticos
atapia@docentes.frgp.utn.edu.ar
- Jeremias Roldan Zocco - Ayudante de trabajos prácticos
jeremias.roldan@alumnos.frgp.utn.edu.ar

Contenidos

- **Vectores y matrices**
- **Asignación dinámica de memoria**
- **Prog. Orientada a objetos**
 - **Encapsulamiento**
 - **Constructores y destructores**
 - **Herencia y composición**
- **Archivos**

Evaluaciones



Examen Integrador

- Trabajo práctico de desarrollo grupal (hasta cuatro personas)
- Defensa grupal con preguntas y/o modificaciones individuales

Software



Codeblocks 20.03



Alternativas:

Zinjai

Dev C++

Visual Studio

Visual Studio Code + plugin C++

Xcode

GitHub



Breve repaso de Laboratorio I

Clasificación de variables

- **Por tipo de dato**

- **Por dimensión**

- **Por alcance**



Clasificación de variables

Por tipo de dato

- **Números enteros**

- **Números reales**

- **Texto**

- **Punteros**



Clasificación de variables

Por tipo de dato

Números enteros: bool, short, int, long

Números reales: float

Texto: char, string

Punteros



Clasificación de variables

Por dimensión

 **Variables simples**

 **Vectores**

 **Matrices**

 **Objetos**



Clasificación de variables: Por dimensión

Variables simples

```
int edad;  
char caracter;  
float precio;  
bool confirmar;
```

Vectores

```
int cant[50];  
char frase[100];  
float recaudacion2020[12];  
bool vec[5];
```

Matrices

```
int notas[50][6];  
char nombres[100][41];  
float ingresos[12][5];  
bool confirmar[3][3];
```



Clasificación de variables

Por alcance

 **Local**

 **Global**



Clasificación de variables: Por alcance

Global

```
int numero;

int main(){
    numero=5;
    return 0;
}

void miFuncion(){
    numero = 10;
    return;
}
```

Local

```
void miFuncion(float var3){
    int var;
    char var2;
    return;
}

int main(){
    int var;
    return 0;
}
```



Breve repaso de Laboratorio I

Estructuras de programación

 **Secuencial**


 **Decisión**

 **Repetición**



Estructuras de programación

 **Decisión:** if ... else, switch

 **Repetición:** for, while, do while



Punteros

```
int main(){  
    int z;  
    z = 5;  
  
    int *p;  
    p = &z;  
    *p = 10;  
    cout << *p;  
    return 0;  
}
```

Se declara una variable llamada p como puntero a entero.

Se asigna la dirección de memoria de la variable z al puntero p. Se puede decir que p "apunta a" z.

En la parte de memoria apuntada por p (o sea el contenido z) se asigna el valor 10.

Se muestra por pantalla el contenido de donde está apuntando p. O sea, el valor de z.



Funciones

```
int maximo(int a, int b){  
    int c;  
    if (a > b){  
        c=a;  
    }  
    else{  
        c=b;  
    }  
    return c;  
}
```

Una función debe indicar un tipo de dato de devolución. **void** si no devuelve nada.

Debe tener un nombre.

Puede o no recibir parámetros.

La palabra clave **return** finaliza la función inmediatamente devolviendo el valor que se indique.



Pasaje de parámetros a funciones

Por valor

```
void mi_funcion ( int param_por_valor );
```

Por dirección

```
void mi_funcion ( int *param_por_direccion );
```

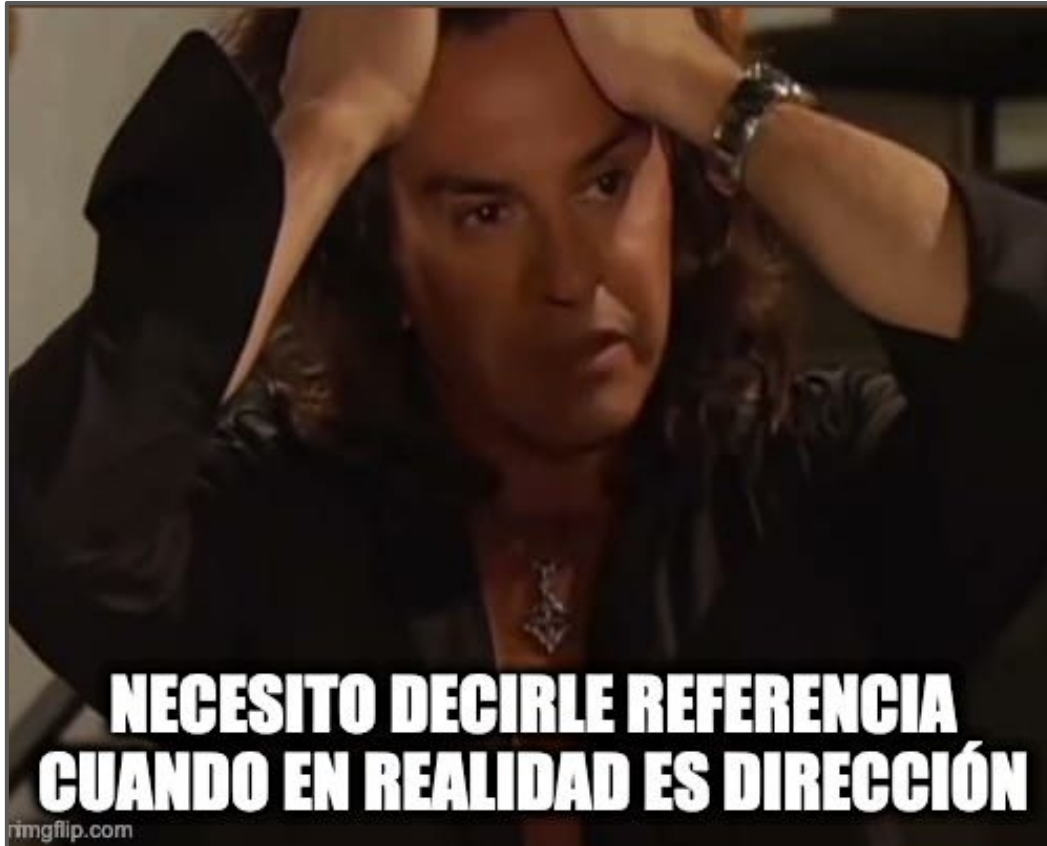
Por referencia

```
void mi_funcion ( int &param_por_referencia );
```

Por favor, no le digan **referencia** cuando en realidad es **por dirección**.



Pasaje de parámetros a funciones



Por favor, no le digan **referencia** cuando en realidad es **por dirección**.

Ejercicio

A un grupo de personas candidatas a presidente se les asignó un número correlativo comenzando desde el 1. En total son siete candidatos.

Se desea un programa que compute los votos totales que obtuvieron los candidatos por municipio. Por cada municipio y candidato se registró:

- Código de municipio (1 a 280)
- Código de candidato (1 a 7)
- Cantidad de votos

No necesariamente existen votos para todos los municipios ni para todos los candidatos. La información no está agrupada ni ordenada. Se indica el fin de la carga de datos con un Código de municipio negativo. Calcular:

- El candidato que más votos ha obtenido. Indicar la cantidad de votos.
- La cantidad de municipios que hayan registrado más de 5000 votos en total.

Ejercicio a resolver

Guía de Trabajos Prácticos Nº1

Ejercicio 5

Una tienda online dispone de todas las ventas realizadas el mes pasado en los tres países donde comercializa. Por cada venta registra:

- Número de país (1 a 3)
- Día de la venta (1 a 31)
- Monto de la venta (float)

Para indicar el fin de la carga de información se ingresa un número de país igual a -1. La información no se encuentra agrupada ni ordenada bajo ningún criterio. Se pide calcular e informar:

- A) Por cada país y día, el total recaudado. Sólo mostrar información de aquellos días en los que hubo recaudación.
- B) Por cada país, la cantidad de días en los que no hubo recaudación.
- C) Los números de día en los que no hubo recaudación en ninguno de los países.

Ejercicio 05 - Trabajo Práctico 01

País	Día	Monto
1	3	500
1	3	400
3	1	600
2	4	700
2	4	800
3	5	900
2	6	300
1	3	400
2	7	200
-1		

 **Carga de datos**

 **Punto A**

 **Punto B**

 **Punto C**

Ejercicio 05 - Trabajo Práctico 01

País	Día	Monto
1	3	500
1	3	400
3	1	600
2	4	700
2	4	800
3	5	900
2	6	300
1	3	400
2	7	200
-1		

Carga de datos: Se ingresan indeterminados registros hasta que se ingrese un número de país igual a -1.

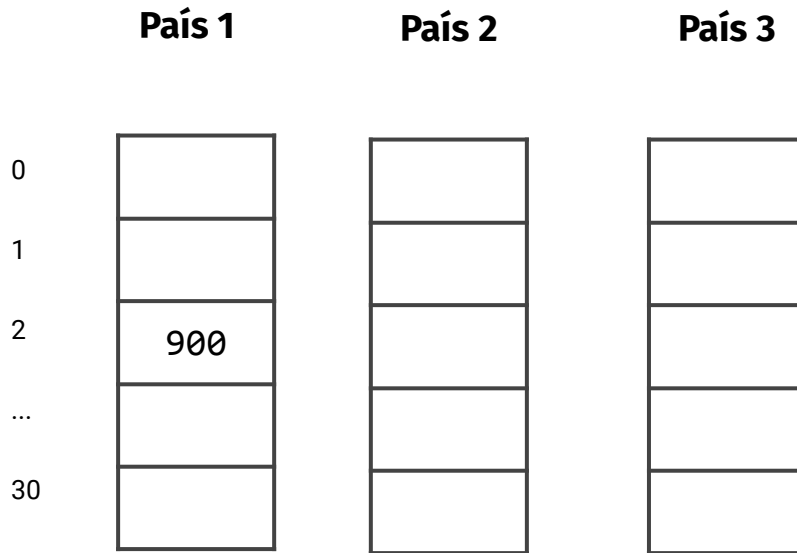
Punto A: Es necesario acumular para cada país los montos de todas las ventas y discriminarlos por día. 3 países y 31 días.

Punto B: Para cada país es necesario identificar y contar los días que no se registró ninguna venta.

Punto C: Por cada día es necesario verificar si ninguno de los 3 países registraron ventas. En ese caso listar el número de día.

Ejercicio 05 - Trabajo Práctico 01

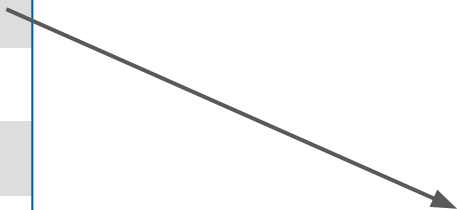
País	Día	Monto
1	3	500
1	3	400
3	1	600
2	4	700
2	4	800
3	5	900
2	6	300
1	3	400
2	7	200
-1		



Resolución utilizando 3 vectores de float de 31

Ejercicio 05 - Trabajo Práctico 01

País	Día	Monto
1	3	500
1	3	400
3	1	600
2	4	700
2	4	800
3	5	900
2	6	300
1	3	400
2	7	200
-1		



	País 1	País 2	País 3
	0	1	2
0			
1			
2	900		
...			
30			

Resolución utilizando una matriz de float 31 x 3