

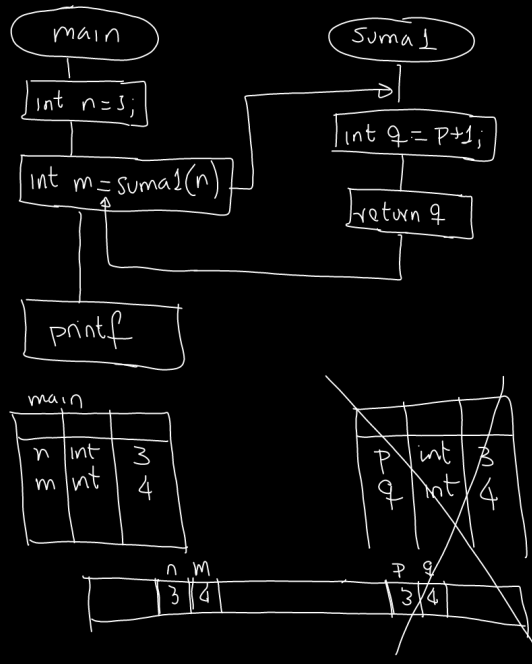
(Versión de fecha 21 de noviembre de 2024)

## 1 REPASO VARIABLES LOCALES

Paso por valor

```
int suma1(int p) {
    int q = p + 1;
    return q;
}

int main() {
    int n = 3;
    int m = suma1(n);
    printf("%d\n", m);
}
```



int, double, char, struct

- Pasan como parámetro por valor
- Utilizan el operador de asignación.
- Se pueden utilizar como valor devuelto.

arrays

- Pasan por referencia.
- No pueden usar asignación.
- No se pueden utilizar como valor devuelto.

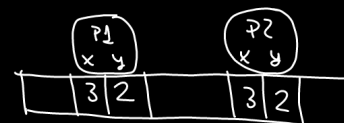
OP. ASIGNACIÓN

```
int x = 2;
int y = x;
```

x = 3;

```
typedef struct {
    int x, y;
} P2D;
```

```
P2D p1;
p1.x = 3; p1.y = 2;
P2D p2 = p1;
```



## Paso de parámetros por referencia

```
void suma2 (int* p) {
```

```
    *p = *p + 2;
```

```
}
```

```
int main() {
```

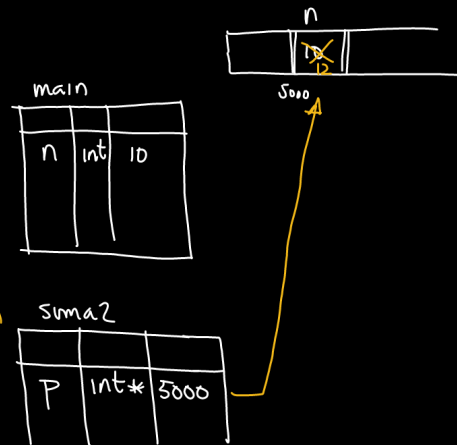
```
    int n = 10;
```

```
    suma2(&n);
```

```
    printf("%d\n", n); // 12
```

```
}
```

→ Paso referencia an  
Paso la dir. de mem de n



El nombre de un array equivale a un puntero al primer elemento.

```
int x[] = {1, 2, 3};
```

```
x ≡ &x[0]
```

```
int suma_elementos(int x[]);
```

```
int suma_elementos(int* x);
```

OP. ASIGNACIÓN: ¡NO!

```
int x[] = {1, 2, 3};
```

```
int y[] = x;
```

```
int* y = x;
```

Nota: los elementos individuales de un array se comportan de acuerdo a su tipo de datos

```
int suma(int x1, int x2) {  
    return (x1+x2);  
}  
  
int main() {  
    int p[] = {1, 2};  
    int resul = suma(p[0], p[1]);  
    p[0] = p[1] + 5;  
}
```

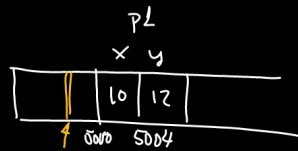
Arrays multi-dimensionales.

int x[] = {1, 2, 3}; <sup>No es necesario</sup> → ¡0! → No vale int x[3];  
x = {1, 2, 3};

int y[][3] = {  
 {0, 1, 0},  
 {1, 0, 1},  
 {-1, -1, 0}  
};

## Punteros a estructuras

```
typedef struct {  
    int x;  
    int y;  
} Punto;
```



```
Punto p1;  
p1.x = 10; p1.y = 12;  
int* ptr_x = &(p1.x); // ptr_x es 5000  
Punto* ptr = &p1;  
(*ptr).x campo x  $\longleftrightarrow$  ptr -> x
```

variable Punto  $\rightarrow$  campo con nomenclatura punto

variable puntero a estruc  $\rightarrow$  nomenclatura flecha