

El test debe ser contestado en la **hoja de lectura óptica**. Sólo una de las cuatro respuestas posibles de cada pregunta es correcta.

El test es eliminatorio y aporta un 30% de la nota final. Son necesarias 8 preguntas correctas (6 con las prácticas aprobadas) para que se corrija el ejercicio.

Cada respuesta correcta: 1 punto. Respuesta incorrecta o en blanco: 0 puntos.

- En el paso de argumentos por valor se pueden utilizar:
 - Sólo variables
 - Sólo variables o expresiones
 - Sólo constantes o expresiones
 - Variables, constantes o expresiones
- El siguiente fragmento de código en C±:


```
x = 15;
do{
    printf("%d ", x);
    x = x/3+1*2;
}while(x%3 == 1);
```

 - Imprime: 15
 - Imprime: 15 7 4
 - Imprime: %15
 - Imprime: 15 7 4 1
- En C±, la sentencia continúe:
 - Finaliza la iteración en curso de cualquier tipo de bucle
 - Finaliza cualquier tipo de bucle
 - Continúa la iteración en curso de cualquier tipo de bucle
 - Continúa cualquier tipo de sentencia
- Al ejecutar el siguiente programa en C±:


```
#include <stdio.h>
typedef struct pp {
    int c1; int c2;
};
int fun(pp x){
    return(x.c1+x.c2);
}
int main(){
    pp val = {4,7};
    printf("%d", fun(val));
}
```

 - Se muestra un valor basura
 - Se muestra el valor 7
 - Se muestra el valor 4
 - Se muestra el valor 11
- Cuando un subprograma hace una llamada a sí mismo:
 - Se dice que es un subprograma compacto
 - Se dice que es un subprograma ascendente
 - Se dice que es un subprograma recursivo
 - Se dice que es un subprograma dicotómico
- En C±, la directiva #pragma once se utiliza:
 - Sólo en los ficheros de interfaz
 - Sólo en los ficheros de implementación
 - Tanto en los ficheros de interfaz como en los de implementación
 - Ni en los ficheros de interfaz ni en los de implementación
- Un procedimiento es puro si:
 - No usa variables externas ni llama a otros subprogramas que no sean puros
 - No usa variables externas pero sí puede llamar a cualquier subprograma sea puro o no
 - Usa variables externas pero no llama a otros subprogramas que no sean puros
 - Usa variables externas y puede llamar a cualquier subprograma sea puro o no
- En el lenguaje C±, la palabra clave private:
 - Oculto elementos de un union
 - Oculto elementos de cualquier estructura de datos
 - Oculto elementos de un struct
 - Oculto elementos de un vector
- Se denomina programa objeto a:
 - La representación de un programa en lenguaje simbólico
 - La representación de un programa en código de máquina
 - Los datos que devuelve un programa ejecutable
 - A los lenguajes de programación orientada a objetos
- Supongamos el fragmento de código en C±:


```
int a, b;
void Suma(int x, int & y) {
    y = y+1;
    y = y+x;
    x = x+1;
}
...
a = 3; b = 5;
Suma(a,b);
```

 Después de la ejecución de Suma(a,b):
 - La variable a vale 4 y la variable b vale 9
 - La variable a vale 4 y la variable b vale 8
 - La variable a vale 3 y la variable b vale 5
 - La variable a vale 3 y la variable b vale 9

EJERCICIO DE PROGRAMACIÓN

Realizar en C± el TAD **ListaCompra** para guardar los datos de una compra de hasta 10 productos diferentes. La compra de cada producto se identifica por el código de artículo (entero), unidades del producto (entero), precio unitario en euros (real) y el porcentaje de IVA (real). La operación **NuevaCompra** pone a cero todos los campos de los 10 productos. La operación **CambiarUnidades**, dado un código de artículo distinto de cero y un nuevo número de unidades mayor que cero, cambia el campo de unidades de ese artículo. La operación **TotalCompra** devuelve el precio total de la compra.