



**ALUMNO:**

Asignatura: Programación I (Grado de Ingeniería Informática – 1<sup>er</sup> curso ).

Curso: **2019/2020**

Examen: Simulacro de Ordinaria

Fecha: 20-12-19

Semestre: Primero

Convocatoria: -

**Teoría: Se debe entregar en papel, con las respuestas correctas (2 puntos)**

**Pregunta 1 ( 1 punto )**

```
#include <iostream>
#include <vector>
#include <functional>

int findus(std::vector<int> const & v, std::function<bool (int, int)> const &
foo){
    int result{0};
    for(int i{0}; i<v.size(); i++){
        if(foo(i,v.at(i))) result += v.at(i);
    }
    return result;
}

int main() {
    int aux = findus({1,2,3,4,5,6}, [](int a, int b)-> bool {
        return ((a+b)%2 == 0);
    });
    std::cout << aux << "\n";
    return 0;
}
```

**¿Qué mostrará el programa por pantalla?**

--



**Pregunta 2: (0.5 puntos)**

```
#include <iostream>
#include <vector>
#include <functional>

int main(){
    int * a = new int;
    int * b;
    int * c = new int;
    *c = 4;
    *a = *c;
    b = a;
    *a = 5;
    std::cout << "a: " << *a << "\n";
    std::cout << "b: " << *b << "\n";
    std::cout << "c: " << *c << "\n";

    // delete instructions are omitted on purpose to avoid giving clues
    return 1;
}
```

**¿Qué mostrará el programa por pantalla?**

**Pregunta 3: (0.5 puntos)**

```
#include <iostream>
#include <vector>
#include <functional>
int aries(int a){
    if(a % 5 == 0) return 5;
    return a+aries(a-1);
}

int main(){
    std::cout << aries(7) << "\n";
    return 0;
}
```

**¿Qué mostrará el programa por pantalla?**

**Se debe entregar 1 único archivo main.cpp a través del Campus Virtual**

### Ejercicio 1 (8 puntos)

Programa que gestione una biblioteca.

- Crear el tipo de dato necesario para un libro (autor, título, tags, ubicación, índice) (2 puntos)
  - Los tags son un conjunto de etiquetas, tipo: “matematicas, primaria, operaciones”.
  - La ubicación es una combinación de letras y números.
  - El índice es un número entero positivo.  
(0.5 puntos)
- Generar menú para gestión biblioteca: (2 puntos)
  - Dar de alta un libro (añadiendo todos sus campos)
  - Dar de baja un libro
  - Listar libros de la biblioteca
  - Modificar los datos de un libro a partir de su índice (el usuario elige el libro por índice y modifica los datos).
  - Buscar un libro (se imprimirán los datos del libro):
    - Por uno de los tags
    - Por título (no tiene por qué estar el título completo, basta que coincidan algunas palabras o caracteres)
    - Por autor (no tiene por qué estar el autor completo, basta que coincidan algunas palabras o caracteres)
- Generar las funciones necesarias para los puntos del apartado anterior (4 puntos - 75% si se usan funciones clásicas, 100% si se usa programación funcional con composición)
  - Dar de alta un libro
  - Dar de baja un libro
  - Listar libros de la biblioteca
  - Modificar los datos de un libro indicado por el usuario
  - Buscar un libro:
    - Por uno de los tags (completo)
    - Por título (incompleto)
    - Por autor.

### Rúbrica de Evaluación

<u>El programa no compila</u> o no se asemeja a lo que se pide	0%
El programa no hace lo que se pide pero el código es correcto y se aproxima a lo pedido	40%
El programa funciona correctamente	60%
Los tipos de datos elegidos son los más adecuados	+20%
Las estrategias y elementos de código elegidos son los más adecuados al problema a resolver	+20%