EXAMEN DE LENGUAJES DE PROGRAMACIÓN II

PREGUNTA 1 (5 puntos)

Responde con la mayor brevedad posible a las siguientes preguntas:

a) Explica tres usos de la palabra **static** en el lenguaje C++.
b) ¿Qué diferencia existe entre **struct** y **union**?
c) Para pasar por referencia una variable en el lenguaje C se define el parámetro correspondiente como un puntero y se pasa la variable en la llamada a la función mediante el operador **&**. ¿Por qué esto no se hace así cuando se trata de vectores?
d) ¿Por qué puede ser necesario emplear miembros protegidos (**protected**) dentro de una clase?
e) ¿Por qué se definen *clases base virtuales*?
f) ¿Qué significa la palabra reservada **this**?
g) ¿Para qué sirven los operadores de conversión?
h) ¿Cuál es la ventaja fundamental que aporta la genericidad?
i) En la entrada/salida, ¿cuál es la clase que aporta los miembros necesarios para el control de errores?
j) ¿Qué instrucción permite lanzar una excepción?
k) Queremos escribir un programa que almacene información sobre las especies animales. Entre las muchas clases que definiremos están las siguientes: **gato, león, mamífero, animal, lista de animales**. Explica la relación que habrá entre estas 5 clases.
l) ¿Qué diferencia existe entre definir una herencia como *pública* o *privada*?
m) ¿Qué relación crees que existe entre la *sobrecarga de funciones* y *las funciones genéricas*?

NOTA: La respuesta correcta de cada pregunta vale 0,4 puntos, a excepción de las preguntas f y g que valen 0,3 puntos.

PREGUNTA 2 (3 puntos)

Dada una clase como la siguiente:

```
class A
{
  int i;
  public:
  .....
    }
```

a) ¿Cómo sería la sobrecarga del operador <<? (cout << a)
b) Define la suma de dos objetos de la clase A. (a+b)
c) Define la suma de un entero y un objeto de la clase A. (a+5; 5+a)
d) Define el incremento de un objeto A. (++a; a++)
e) Es necesario definir el constructor copia o el operador

de asignación. Indica cuál o cuáles son necesarios y razona tu respuesta.

PREGUNTA 3. (2 Puntos)

Añade las clases y funciones necesarias para que el siguiente programa sea compilable y ejecutable.

```
void main ( )
{
    A *a = NULL;
    B b ( 5 );
    C c ( "hola" );

    Insertar ( &a, &b );
    Insertar ( &a, &c );
    Mostrar ( a );
}
```

a) No se debe modificar ni añadir nada a la función que se proporciona.
b) En las clases que se definan no será posible añadir ningún campo cuyo contenido tenga como finalidad identificar a la clase.
c) La salida en pantalla, que produce la función **Mostrar,** debe ser la siguiente:
hola5

n) Sólo se escribirá una función **Insertar.**
o) El puntero **a** es la cabeza de una