

Pag. 1 Fecha: 15/MAR/2012 Hora: 13:24 hrs.

INSTITUTO TECNOLOGICO DE NUEVO LAREDO

ING. EN SISTEMAS COMPUTACIONALES

ING. BRUNO LOPEZ TAKEYAS

<<< P R E G U N T A S >>>

Unidad	Folio	Pregunta
=====		

POOC#	PROGRAMACION ORIENTADA A OBJETOS EN C#	
3 .	1	En C# se permite la herencia simple a) Verdadero b) Falso
3 .	2	En C# se permite la herencia multiple a) Verdadero b) Falso
3 .	3	Los miembros de una clase base SOLAMENTE se heredan a las clases derivadas directas; es decir, no se heredan a las clases "hijas" de sus subclases a) Verdadero b) Falso
3 .	4	Caracteristica de POO que permite definir nuevas clases a partir de otras ya existentes a) Herencia b) Abstraccion c) Encapsulamiento d) Constructor
3 .	5	Suponga que tiene 2 clases A y B, donde B hereda de la clase A, Como se declara la clase B? a) class A : B b) Class A : B c) class B : A d) Class B : A
3 .	6	Si se tiene la siguiente declaracion: class Empleado : Persona { } Se entiende que ... a) La clase Persona hereda de la clase Empleado b) La clase Empleado hereda de la clase Persona c) El objeto Empleado hereda de la clase Persona d) El objeto Empleado hereda del objeto Persona
3 .	7	Si se tiene la siguiente declaracion: class Empleado : Persona { } Se entiende que ... a) La clase Persona es derivada de la clase Empleado b) La clase hija es Persona c) La clase base es Empleado d) La clase Empleado es derivada de la clase Persona
3 .	8	Si se tiene la siguiente declaracion: class Empleado : Persona { } Se entiende que ... a) La clase base es Empleado b) La subclase es Persona c) La clase base es Persona d) Todas las anteriores

- 3 . 9 Si se tiene la siguiente declaracion:
class Empleado : Persona
{
....
}
Se entiende que ...
a) La clase base es Persona
b) La subclase es Empleado
c) La clase derivada es Empleado
d) Todas las anteriores
- 3 . 10 Si se tiene la siguiente declaracion:
class Empleado : Persona
{
....
}
Se entiende que ...
a) La superclase es Persona
b) La subclase es Empleado
c) La clase derivada es Empleado
d) Todas las anteriores
- 3 . 11 Si se tiene la siguiente declaracion:
class Empleado : Persona
{
....
}
Se entiende que ...
a) La superclase es Empleado
b) La subclase es Persona
c) La superclase es Persona
d) Todas las anteriores
- 3 . 12 Una subclase puede llamar los metodos de su superclase
a) Verdadero
b) Falso
- 3 . 13 Si la clase C hereda de B y la clase B hereda de A, se dice que la clase C forma parte de una jerarquia de ...

(Nota: No hay mas clases involucradas en la jerarquia)
a) Herencia publica
b) Herencia multiple
c) Herencia simple
d) Herencia simple y multiple
- 3 . 14 Al implementar herencia, se recomienda que los elementos comunes se coloquen en la superclase
a) Verdadero
b) Falso
- 3 . 15 Al implementar herencia, se recomienda que los elementos comunes se coloquen en las clases derivadas
a) Verdadero
b) Falso
- 3 . 16 Al implementar herencia, se recomienda que los elementos comunes se coloquen en la clase base
a) Verdadero
b) Falso
- 3 . 17 Los elementos heredados por una clase derivada se acceden igual que si estuvieran presentes en ella
a) Verdadero
b) Falso
- 3 . 18 Si se tiene el siguiente fragmento de codigo:
class A {
public void Cambiar() { ... }
}
class B : A { ...}
En la clase B, Como se invoca el metodo Cambiar() de la clase base desde la clase derivada ?
a) A.Cambiar();
b) B.Cambiar();
c) base.Cambiar();
d) No se puede invocar
- 3 . 19 Se puede invocar el constructor de la clase base desde el constructor de una clase derivada
a) Verdadero
b) Falso

- 3 . 20 Para invocar un metodo llamado Consultar() que se encuentra en la clase base se utiliza ...
a) base.Consultar();
b) :base.Consultar();
c) base.Consultar;
d) :base(Consultar());
- 3 . 21 Al observar la siguiente instruccion
:base(nombre, direccion)
resulta evidente que este se encuentra ubicada en ...
a) Una propiedad
b) Un metodo cualquiera
c) Un metodo destructor
d) Un metodo constructor
- 3 . 22 Para invocar un metodo llamado Consultar() que se encuentra en una clase hija, se utiliza ...
a) base.Consultar();
b) :base.Consultar();
c) :base(Consultar());
d) Ninguna de las anteriores
- 3 . 23 Si se tiene la clase A con su respectivo constructor de fault y la clase B hereda de A, Como se invoca el constructor de la clase A desde el constructor de la clase B ?
a) public B(): base()
b) public B(): A.A();
c) public A(): base();
d) public A(): B.A();
- 3 . 24 Cuando el constructor de una clase derivada invoca al constructor de su clase base, primero se ejecuta el constructor de la clase base y luego el constructor de la clase derivada
a) Verdadero
b) Falso
- 3 . 25 Cuando el constructor de una clase derivada invoca al constructor de su clase base, primero se ejecuta el constructor de la clase derivada y luego el constructor la clase base
a) Verdadero
b) Falso
- 3 . 26 Tipo de relacion entre clases que se utiliza para modelar un "todo" y sus "partes", donde el "todo" no puede existir si no existen sus "partes"
a) Composicion
b) Agregacion
c) Herencia
d) Cardinalidad
- 3 . 27 Tipo de relacion entre clases que se utiliza para modelar un "todo" y sus "partes", donde las "partes" desaparecen cuando se elimina el "todo"
a) Composicion
b) Agregacion
c) Herencia simple
d) Herencia multiple
- 3 . 28 Tipo de relacion entre clases que se representa mediante un rombo negro en los diagramas de UML
a) Composicion
b) Agregacion
c) Herencia simple
d) Herencia multiple
- 3 . 29 Tipo de relacion entre clases que se representa mediante un rombo blanco en los diagramas de UML
a) Composicion
b) Agregacion
c) Herencia simple
d) Herencia multiple
- 3 . 30 Cuando un objeto se encuentra compuesto por otros objetos, se dice que existe ...
a) Composicion
b) Agregacion

- c) Herencia
 - d) Cardinalidad
- 3 . 31 Tipo de conexion entre clases que permite implementar relaciones de tipo "tiene un"
- a) Composicion
 - b) Agregacion
 - c) Herencia
 - d) Cardinalidad
- 3 . 32 Tipo de conexion entre clases que permite implementar relaciones de tipo "es un"
- a) Composicion
 - b) Agregacion
 - c) Herencia
 - d) Cardinalidad
- 3 . 33 Cuando se desea incorporar en una clase las variables, propiedades y metodos de otra clase, entonces se implementa ...
- a) Composicion
 - b) Agregacion
 - c) Herencia
 - d) Cardinalidad
- 3 . 34 Cuando se desea que un objeto contenga otro objeto, entonces se implementa ...
- a) Composicion
 - b) Agregacion
 - c) Herencia
 - d) Cardinalidad
- 3 . 35 Tipo de relacion entre clases que se utiliza para modelar un "todo" y sus "partes", donde el "todo" se forma incorporando sus "partes"
- a) Composicion
 - b) Agregacion
 - c) Herencia
 - d) Cardinalidad
- 3 . 36 Tipo de relacion entre clases que se utiliza para modelar un "todo" y sus "partes", donde las partes pueden pertenecer a varios "todos"
- a) Composicion
 - b) Agregacion
 - c) Herencia
 - d) Cardinalidad
- 3 . 37 Tipo de relacion entre clases que se utiliza para modelar un "todo" y sus "partes", donde la existencia de objetos de dichas clases es independiente
- a) Composicion
 - b) Agregacion
 - c) Herencia
 - d) Cardinalidad
- 3 . 38 Tipo de relacion entre clases que se utiliza para modelar un "todo" y sus "partes", en la que si se elimina el "todo", pueden seguir existiendo sus "partes"
- a) Composicion
 - b) Agregacion
 - c) Herencia
 - d) Cardinalidad
- 3 . 39 Una clase sellada (sealed) puede ser heredada
- a) Verdadero
 - b) Falso
- 3 . 40 Una clase sellada (sealed) puede ser instanciada
- a) Verdadero
 - b) Falso
- 3 . 41 Una clase sellada (sealed) se utiliza para evitar la herencia
- a) Verdadero
 - b) Falso
- 3 . 42 Una clase sellada (sealed) representa lo mismo que una clase abstracta
- a) Verdadero
 - b) Falso

- 3 . 43 Si se usa simultaneamente una clase sealed con una clase abstracta, entonces se produce un error
a) Verdadero
b) Falso
- 3 . 44 Se puede usar simultaneamente una clase sealed con una clase abstracta
a) Verdadero
b) Falso
- 3 . 45 En una relacion de composicion, el "todo" puede existir aunque no existan sus "partes"
a) Verdadero
b) Falso
- 3 . 46 En una relacion de composicion, el "todo" no puede existir si no existen sus "partes"
a) Verdadero
b) Falso
- 3 . 47 En una relacion de composicion, sus "partes" desaparecen cuando se elimina el "todo"
a) Verdadero
b) Falso
- 3 . 48 En una relacion de composicion, cuando se elimina el "todo", sus "partes" pueden seguir existiendo
a) Verdadero
b) Falso
- 3 . 49 La relacion de composicion se representa mediante un rombo negro en diagramas de UML
a) Verdadero
b) Falso
- 3 . 50 La relacion de composicion se representa mediante un rombo blanco
a) Verdadero
b) Falso
- 3 . 51 La relacion de agregacion se representa mediante un rombo negro
a) Verdadero
b) Falso
- 3 . 52 La relacion de agregacion se representa mediante un rombo blanco
a) Verdadero
b) Falso
- 3 . 53 La relacion de composicion se presenta cuando un objeto se encuentra compuesto por otros objetos
a) Verdadero
b) Falso
- 3 . 54 La relacion de agregacion se presenta cuando un objeto se encuentra compuesto por otros objetos
a) Verdadero
b) Falso
- 3 . 55 Cuando se presentan relaciones de tipo "tiene un", entonces se implementa composicion
a) Verdadero
b) Falso
- 3 . 56 Cuando se presentan relaciones de tipo "tiene un", entonces se implementa agregacion
a) Verdadero
b) Falso
- 3 . 57 Cuando se presentan relaciones de tipo "tiene un", entonces se implementa herencia
a) Verdadero
b) Falso
- 3 . 58 Cuando se presentan relaciones de tipo "es un", entonces se implementa herencia
a) Verdadero
b) Falso

- 3 . 59 Cuando se desea incorporar en una clase las variables, propiedades y metodos de otra clase, entonces se implementa herencia
a) Verdadero
b) Falso
- 3 . 60 Cuando se desea incorporar en una clase las variables, propiedades y metodos de otra clase, entonces se implementa composicion
a) Verdadero
b) Falso
- 3 . 61 Cuando se desea incorporar en una clase las variables, propiedades y metodos de otra clase, entonces se implementa agregacion
a) Verdadero
b) Falso
- 3 . 62 Cuando se desea que un objeto contenga otro objeto, entonces se implementa la composicion
a) Verdadero
b) Falso
- 3 . 63 En la relacion de agregacion, la existencia de los objetos de las clases involucradas es independiente
a) Verdadero
b) Falso
- 3 . 64 En la relacion de composicion, la existencia de los objetos de las clases involucradas es independiente
a) Verdadero
b) Falso
- 3 . 65 En la relacion de composicion, las "partes" pueden pertenecer a varios "todos"
a) Verdadero
b) Falso
- 3 . 66 En la relacion de agregacion, las "partes" pueden pertenecer a varios "todos"
a) Verdadero
b) Falso
- 3 . 67 En la relacion de agregacion, si se elimina el "todo", pueden seguir existiendo sus "partes"
a) Verdadero
b) Falso
- 3 . 68 En la relacion de composicion, si se elimina el "todo" pueden seguir existiendo sus "partes"
a) Verdadero
b) Falso
- 3 . 69 El objetivo de una clase sealed es evitar la herencia
a) Verdadero
b) Falso
- 3 . 70 Una clase sealed puede heredar de otras clases
a) Verdadero
b) Falso
- 3 . 71 Una clase sealed puede ser una clase base
a) Verdadero
b) Falso
- 3 . 72 Si la clase C hereda simultaneamente de las clases B y A, se dice que la clase C forma parte de una jerarquia de ... (Nota: No hay mas clases involucradas en la jerarquia)
a) Herencia publica
b) Herencia multiple
c) Herencia simple
d) Herencia simple y multiple
- 3 . 73 Cual es el namespace que contiene las clases e interfaces que permiten modelar estructuras de datos
a) System.Windows.Forms
b) System.Text
c) System.Collection.Generic
d) System.Console

- 3 . 74 Coleccion generica que permite modelar arreglos
a) ArrayList
b) Stack
c) Queue
d) List
- 3 . 75 Coleccion generica que permite modelar pilas
a) ArrayList
b) Stack
c) Queue
d) List
- 3 . 76 Coleccion generica que permite modelar colas
a) ArrayList
b) Stack
c) Queue
d) List
- 3 . 77 Coleccion generica que permite modelar listas simples
a) Queue
b) Stack
c) List
d) LinkedList
- 3 . 78 Coleccion generica que permite modelar listas dobles
a) Queue
b) Stack
c) List
d) LinkedList
- 3 . 79 Si se tiene una clase llamada Alumno. La siguiente instruccion se utiliza para crear una lista generica de objetos de tipo Alumno
a) List<Alumnos> lista = new List<Alumnos>;
b) List<Alumno> lista = new List<Alumno>;
c) List(Alumno) lista = new List(Alumno);
d) List<Alumno> lista = new List<Alumno>();
- 3 . 80 El metodo Clear() de la coleccion generica List elimina un nodo particular de la lista
a) Verdadero
b) Falso
- 3 . 81 Metodo para insertar un nodo en una lista generica
a) Clear
b) Add
c) Insert
d) Agregate
- 3 . 82 Metodo para eliminar un nodo de una lista generica
a) Remove
b) Clear
c) Delete
d) RemoveAll
- 3 . 83 Metodo para eliminar todos los nodos de una lista generica
a) Remove
b) Clear
c) Delete
d) RemoveAll
- 3 . 84 Un iterador es un conjunto de instrucciones que devuelve una secuencia de valores de un mismo tipo
a) Verdadero
b) Falso
- 3 . 85 Un iterador puede invocarse mediante un ciclo foreach
a) Verdadero
b) Falso
- 3 . 86 Si se declara...
List<Alumno> Lista = new List<Alumno>();
Como se invoca su iterador mediante un ciclo foreach ?
a) foreach(int Alumno in Lista)
b) foreach(Alumno x in Lista)
c) foreach(Alumno x in Lista.Alumno)
d) foreach(Lista x in Alumno)
- 3 . 87 Como se implementa un ciclo foreach para recorrer todos

- los controles de un groupBox1 ?
- a) foreach(Control c in groupBox1)
 - b) foreach(Control c in groupBox1.Controls)
 - c) foreach(Control c in groupBox1.Forms)
 - d) foreach(Control c in groupBox1.Controls)
- 3 . 88 El metodo GetEnumerator() sirve para implementar un iterador
- a) Verdadero
 - b) Falso
- 3 . 89 El metodo GetEnumerator() devuelve un enumerador
- a) Verdadero
 - b) Falso
- 3 . 90 Sentencias para devolver un valor de un iterador
- a) return
 - b) yield return
 - c) yield
 - d) yield break
- 3 . 91 Sentencias para finalizar los ciclos de un iterador
- a) return
 - b) yield return
 - c) yield
 - d) yield break
- 3 . 92 Identifique la(s) relacion(es) en el sig. codigo:
- ```
class A : B {
 public A() {

 }
}
```
- a) Herencia
  - b) Composicion
  - c) Agregacion
  - d) Ninguna de las anteriores
- 3 . 93 Identifique la(s) relacion(es) en el sig. codigo:
- ```
class A {  
    public A() {  
        ....  
    }  
    ~A() { ... }  
}
```
- a) Herencia
 - b) Composicion
 - c) Agregacion
 - d) Ninguna de las anteriores
- 3 . 94 Identifique la(s) relacion(es) en el sig. codigo:
- ```
class A {
 private B x = new B();

 public void Insertar(B x) { ... }
 ~A() { ... }
}
```
- a) Herencia
  - b) Composicion
  - c) Agregacion
  - d) Ninguna de las anteriores
- 3 . 95 Identifique la(s) relacion(es) en el sig. codigo:
- ```
class A {  
    private List<B> x = new List<B>();  
    ....  
    public void Insertar(B x) { ... }  
    ~A() { ... }  
}
```
- a) Herencia
 - b) Composicion
 - c) Agregacion
 - d) Ninguna de las anteriores
- 3 . 96 Identifique la(s) relacion(es) en el sig. codigo:
- ```
class A {
 private List x = new List();

 public void Insertar(B x) { ... }
}
```
- a) Herencia



- b) Composicion  
c) Agregacion  
d) Ninguna de las anteriores
- 3 . 97 Identifique la(s) relacion(es) en el sig. codigo:
- ```
class A {  
    private B x = new B();  
    ....  
    public void Insertar(B x) { ... }  
}
```
- a) Herencia
b) Composicion
c) Agregacion
d) Ninguna de las anteriores
- 3 . 98 La relacion de composicion entre clases es jerarquica
a) Verdadero
b) Falso
- 3 . 99 La relacion de agregacion entre clases es jerarquica
a) Verdadero
b) Falso
- 3 . 100 La relacion de herencia entre clases es jerarquica
a) Verdadero
b) Falso
- 3 . 101 En una relacion de composicion entre clases, la clase del todo hereda todos los componentes de la clase de sus partes
a) Verdadero
b) Falso
- 3 . 102 En una relacion de agregacion entre clases, la clase del todo hereda todos los componentes de la clase de sus partes
a) Verdadero
b) Falso
- 3 . 103 La coleccion generica Queue modela estructuras de datos de tipo pila
a) Verdadero
b) Falso
- 3 . 104 La coleccion generica Stack modela estructuras de datos de tipo pila
a) Verdadero
b) Falso
- 3 . 105 En relaciones de herencia, primero se ejecuta el constructor de la clase base y despues el constructor de la clase derivada
a) Verdadero
b) Falso
- 3 . 106 En relaciones de herencia, primero se ejecuta el constructor de la clase derivada y despues el constructor clase base
a) Verdadero
b) Falso
- 3 . 107 La coleccion generica LinkedList modela estructuras de datos de tipo lista enlazada simple
a) Verdadero
b) Falso
- 3 . 108 Para modelar una lista doble se usa la coleccion generica List
a) Verdadero
b) Falso

=====

TOTAL: 108 PREGUNTAS