

¿Qué hemos aprendido del módulo I?

11/11 puntos (100.00 %)

Cuestionario, 11 preguntas

✓ **¡Felicitaciones! ¡Aprobaste!**

Próximo artículo



1 / 1
puntos

1.

1. Lee en Wikipedia el texto correspondiente a "Máquina Virtual de Java" (JVM por sus siglas en inglés) en: https://es.wikipedia.org/wiki/M%C3%A1quina_virtual_Java

2. Lee el funcionamiento de intérpretes y compiladores en Wikipedia en: [https://es.wikipedia.org/wiki/Int%C3%A9rprete_\(inform%C3%A1tica\)](https://es.wikipedia.org/wiki/Int%C3%A9rprete_(inform%C3%A1tica))

Con base, en lo leído con anterioridad contesta las siguientes preguntas.

Elige el **principio general** de diseño de la solución a un problema:

- Da el algoritmo que resuelva el problema
- Da una breve descripción del problema
- Divide y vencerás

Correcto

😊 Tanto en programación estructurada (que se tiene que usar en ciertos niveles, aunque estemos usando programación orientada a objetos) como en diseño y programación orientada a objetos, debes identificar las partes en las que puedes dividir o distribuir la solución de un problema.

- Determina las entradas que requiere la solución
- Determina las salidas que debe proporcionar la solución

¿Qué hemos aprendido del módulo I?

1 / 1

puntos

11/11 puntos (100.00 %)

Cuestionario, 11 preguntas

Da el orden en que se deben listar los componentes de la definición de un algoritmo:

- a) Lista de pasos a ejecutar
- b) La ejecución siempre termina con el resultado correcto
- c) Cero o más entradas
- d) Cada paso bien definido
- e) Una o más salidas

 e,d,c,a,b a,b,c,d,e c,e,a,d,b**Correcto**

😊 Este es el orden que se da en la definición de un algoritmo: se empieza por las entradas y se termina hablando de la ejecución.

¿Qué hemos aprendido del módulo I?

1 / 1

Puntos

11/11 puntos (100.00 %)

Cuestionario, 11 preguntas

De los pasos listados en la pregunta 2, ¿cuáles serían relacionados con la **sintaxis** (forma) y cuáles con la **semántica** (ejecución) del algoritmo?

- Sintaxis: a, b
Semántica: c, d, e
- Sintaxis: a, d
Semántica: b, c, e



Correcto

😊 Sintaxis: La lista de pasos a ejecutar, que dice que debe ser finita, tiene que ver con la forma que debe tomar el algoritmo, lo mismo que la especificación de cada paso, pues "bien especificado" tiene que ser de acuerdo a ciertas reglas dadas previamente.

Semántica: Si el algoritmo recibe cero o más entradas tiene que ver con la ejecución, lo mismo que producir salidas. Asimismo, que el algoritmo entregue la respuesta correcta (sea *eficaz*) tiene que ver con cómo transcurra la ejecución.

- Sintaxis: c, e
Semántica: a, b, d
-

¿Qué hemos aprendido del módulo I?



1 / 1
puntos

11/11 puntos (100.00 %)

Cuestionario, 11 preguntas

Elige la descripción que defina de mejor manera al diseño orientado a objetos y que **sólo se aplique** a este último:

- En el diseño orientado a objetos se busca dividir el problema en periodos de tiempo
- En el diseño orientado a objetos se busca dividir el problema en objetivos bien identificados
- En el diseño orientado a objetos se busca dividir el problema en actores y responsabilidades



Correcto

😊 Los actores en el diseño de una solución orientada a objetos son los objetos, quienes se van a encargar de asumir ciertas responsabilidades, a lo que llamamos servicios o métodos, dependiendo de la clase a la que pertenezcan.

- En el diseño orientado a objetos se busca dividir el problema en funciones propias y funciones de biblioteca (*library*)
-

¿Qué hemos aprendido del módulo I?

1 / 1

puntos

11/11 puntos (100.00 %)

Cuestionario, 11 preguntas

Elige la descripción del término *acceso*.

- Define quién puede acceder a la clase o clases del proyecto
- Define la forma en que se usan todos los atributos del objeto
- Define quién puede teclear datos al proyecto
- Define quién puede usar directamente cada uno de los componentes de una clase

**Correcto**

😊 Dependiendo del acceso que se asigne a cada componente (atributo o método) queda definido quienes lo conocen fuera de la clase.



1 / 1

puntos

6.

Elige la mejor descripción de lo que es el *polimorfismo*.

- Cuando una misma acción produce resultados distintos dependiendo de quién la ejecute

**Correcto**

😊 Por ejemplo, la acción de producir sonido tiene resultados distintos si el sonido es el que produce un violín o la garganta de una persona.

- Cuando una misma acción tiene varios sinónimos
- Cuando una clase tiene muchas acciones que puede realizar
- Cuando una variable puede tomar valores distintos

¿Qué hemos aprendido del módulo I?

1 / 1

puntos

11/11 puntos (100.00 %)

Cuestionario, 11 preguntas

Elige la mejor descripción de la relación que hay entre una *clase* y un *objeto* de esa clase.

- Es lo mismo una clase que un objeto
- Las clases describen lo que contienen los objetos y lo que saben hacer

**Correcto**

😊 Una clase es una patrón o plantilla, como los planos de una casa, para construir objetos que tengan la forma que la clase dice y sepan hacer lo que la clase dice que saben hacer.

- Las clases contienen a los objetos de la clase
- Los objetos describen lo que contienen las clases y lo que saben hacer



1 / 1

puntos

8.

Elige la opción que complete el enunciado: Un *mensaje*, en la orientación a objetos es:

- Una solicitud de servicio (una invocación de un método)

**Correcto**

😊 El mensaje consiste del objeto al que va dirigido con la sintaxis exacta que tenga definido el método.

- Una cadena que se le escribe al usuario
- Un atributo de un objeto
- Algo a lo que responde alguna de las clases

¿Qué hemos aprendido del módulo I?



1 / 1
puntos

11/11 puntos (100.00 %)

Cuestionario, 11 preguntas

El *estado* de un objeto está dado por la colección de valores que guardan sus atributos. Si los atributos son la *edad* en años cumplidos, la *estatura* en metros y el color de *ojos*. ¿Cuál de las siguientes opciones describe el estado de un objeto?

(edad=105, estatura=.75, ojos=verde)



Correcto

😊 Aunque la edad nos pueda parecer mucha y la estatura poca, cada atributo tiene un valor del tipo especificado.

(edad=17, estatura = rojo, ojos=1.75)

(ojos=azul, estatura=1.5, edad=viejo)

(17, 12, rojo)

¿Qué hemos aprendido del módulo I?



1 / 1

puntos

11/11 puntos (100.00 %)

Cuestionario, 11 preguntas

Elige 2 opciones que respondan a la siguiente pregunta: ¿Por qué a Java se le reconoce como un lenguaje orientado a objetos?

Porque cuando se usa Java se diseñan las soluciones con orientación a objetos



Deseleccionado es lo correcto

Porque todo está definido dentro de alguna clase



Correcto



La programación en Java se hace definiendo clases y construyendo objetos de esas clases. Todo tiene que estar definido dentro de clases.

Porque las bibliotecas son paquetes de clases con métodos y atributos que pueden ser usados por cualquier otra clase



Correcto



Java es un lenguaje centrado en clases y como las clases son las que describen a los objetos, a Java se le considera un lenguaje orientado a objetos.

Porque al resolver un problema usando Java, participan varios objetos de manera distribuida para resolver un problema



Deseleccionado es lo correcto

¿Qué hemos aprendido del módulo I?



1 / 1
puntos

11/11 puntos (100.00 %)

Cuestionario, 11 preguntas

Elige 2 opciones que completen correctamente el siguiente enunciado: El encapsulamiento en la orientación a objetos consiste en

Todos los objetos de la misma clase están dentro de una cápsula, se pueden ver entre sí pero no pueden ver hacia afuera



Deseleccionado es lo correcto

Cada objeto se encuentra totalmente aislado del resto de los objetos por una cápsula en la que está envuelto



Deseleccionado es lo correcto

Cada objeto está en una cápsula, donde se encuentran sus atributos y métodos y en la que decide qué es lo que muestra de su interior y cuáles solicitudes acepta



Correcto



Este es el principio del encapsulamiento: los atributos o variables están junto a los métodos y la clase decide qué se puede ver desde afuera y qué no.

La clase determina la forma de la cápsula que va a poner alrededor de "sus" objetos



Correcto



Dependiendo del acceso que dé a sus atributos y métodos, cada objeto queda encapsulado, dando o no acceso a los objetos de otras clases.

¿Qué hemos aprendido del módulo I?

11/11 puntos (100.00 %)

Cuestionario   4 preguntas 