



Tecnológico de Estudios Superiores del Oriente del Estado de México

Martínez Martínez Enrique Caciél

Grupo: 2S11

MArteria: Programación orientada a objetos

<https://github.com/caciél1718>

2.4 Declaración de clases

Clase El elemento básico de la programación orientada a objetos en Java es la clase. Una clase define la forma y comportamiento de un objeto.

Los datos. Los datos se encapsulan dentro de una clase declarando variables dentro de las llaves de apertura y cierre de la declaración de la clase variables que se conocen como atributos. Se declaran igual que las variables locales de un método en concreto.

```
{ int X, Y }
```

Métodos Los métodos son subrutinas que definen la interacción de una clase, sus capacidades y comportamientos. Un método ha de tener por método nombre cualquier identificador legal, distinto de los y utilizados por los nombres de las clases en que esta definida.

```
int metodo suma (int paramX, int paramY) {  
    return (paramX + paramY);  
}
```

2.2 Instanciación de una clase

Los tipos simples de Java describen el tamaño y los valores de las clases de las variables. Cada vez que se crea una clase se añade otro tipo de dato que se puede utilizar igual que uno de los tipos simples. Por ello al declarar una nueva variable se puede utilizar un nombre de clase como tipo. A estas variables se las conoce como referencias a objetos.

2.3 Referencia a el objeto actual

[Ejemplo this from command-line caption] hace que el código de programa pueda alternarse entre objetos porque evita el nombre del objeto y encapsula automáticamente la clase primaria del objeto.

2.4 Métodos

declaración, mensaje, paso de parámetros, retorno de valores

Declaración Especifica el tipo de acceso permitido indicando qué espacios de la clase podrán acceder a ese método, los métodos son la única forma de acceso a los atributos privados

Paso de parámetros Los parámetros de una función son variables locales que se inicializan en el momento de la llamada al método. Fuera de la función no se conocen y no pueden ser accedidos

Retorno de valores Un método puede devolver un valor a quien lo llama o no devolver nada. El valor devuelto por un método puede ser de un tipo primitivo de datos o una referencia, pero nunca puede devolver más de un valor

2.5 Constructores y destructores

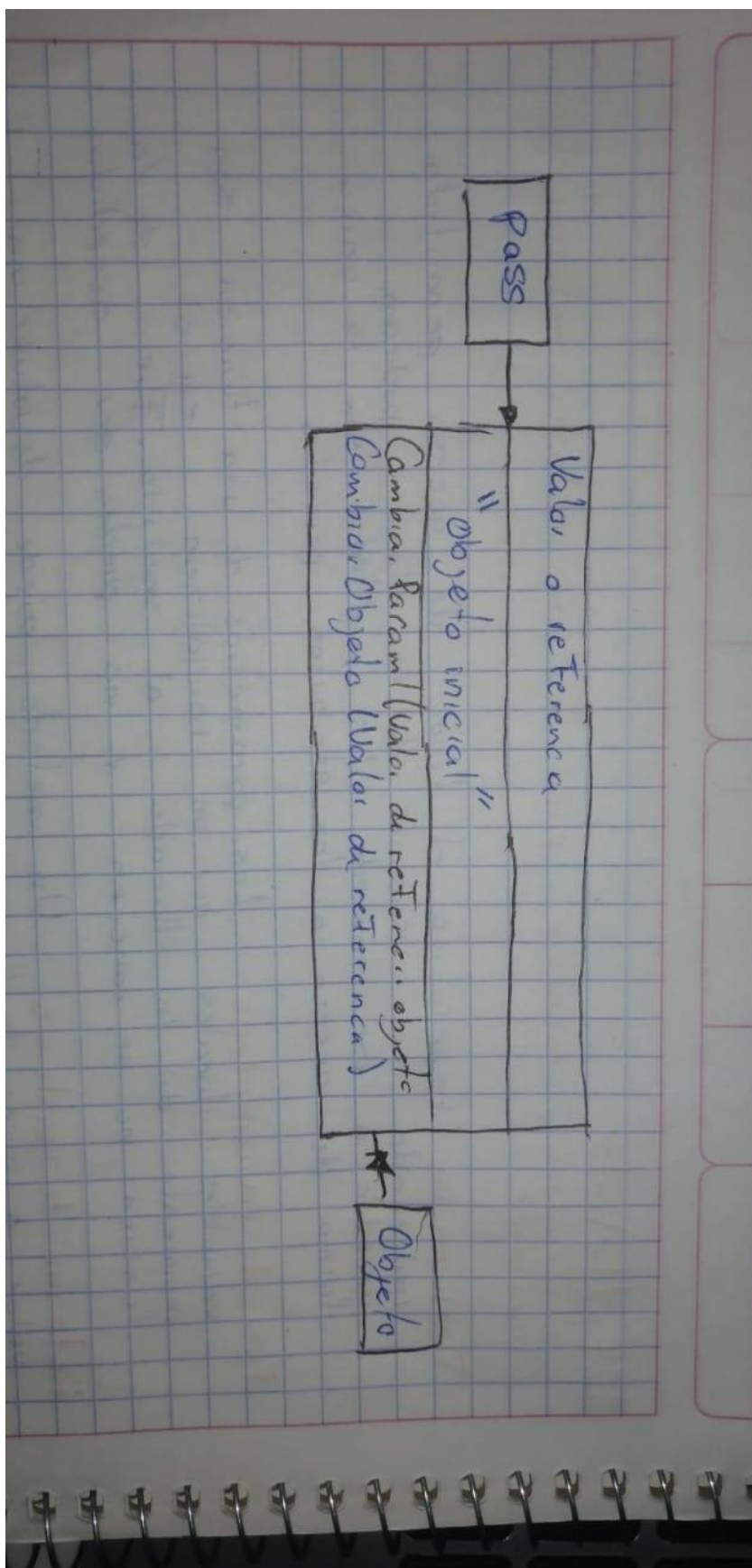
Constructor Es el de inicializar un objeto cuando este es creado. Asignaremos los valores iniciales así como las acciones que esta clase deba realizar

Destructor Es un método de una clase que se llama justo antes de una instancia de esa clase y se eliminan de la memoria. No todos los lenguajes de programación orientados a objetos suelen tener un destructor

2.6 Sobrecarga de métodos

La sobrecarga de métodos se aplica de métodos y constructores, los destructores no se pueden sobrecargar ya que debe haber uno por clase

La sobrecarga de métodos hace que un mismo nombre pueda representar distintos métodos con distintos tipo y número de parámetros



2.7 Sobrecarga de operadores

Los operadores son un tipo de tokens que indican al compilador la realización de determinadas operaciones sobre variables u otros. La sobrecarga de operadores permite redefinir ciertos operadores como "+" y "*", para usarlos con la clase que hemos definido.

La sobrecarga de operadores solo se puede usar con clases, no se puede redefinir los operadores para los tipos simples predefinidos.

La sobrecarga de operadores es la capacidad de transformar los operadores de un lenguaje como por ejemplo el "+", "-", etc. cuando se dice transformar se refiere a que los operadores que entran en juego no tiene que ser lo que admite el lenguaje por defecto. Mediante esta técnica podemos sumar dos objetos creados por nosotros a un objeto y un entero, en vez de limitarnos a sumar números enteros o reales por ejemplo.