

Ejercicios de Programación (Ing. Telecomunicaciones)

Enunciado

¿Cuál de las siguientes sentencias dará error de compilación?

```
int calculaArea(int altura, int anchura) {  
    return altura * anchura;  
}  
  
short c = 7;  
calculaArea(c, 15);  
long t = 42;  
int f = calculaArea(t, 17);  
byte h = calculaArea(4, 20);  
long j = calculaArea(2, 3);
```

Solución

Las líneas siguientes no compilan por incompatibilidad de tipos:

```
int f = calculaArea(t, 17);  
byte h = calculaArea(4, 20);
```

Enunciado

¿Qué imprime en la terminal este programa?

```
public class Ejemplo {  
  
    static class Clase {  
        String a;  
        public Clase(String a) {  
            this.a = a;  
        }  
    }  
  
    static void metodo(int entero, String texto, Clase instancia) {  
        entero++;  
        texto += " HOLA ";  
        instancia.a += " HOLA ";  
        System.out.println("En metodo:");  
        System.out.println(entero);  
        System.out.println(texto);  
        System.out.println(instancia.a);  
    }  
}
```

```

    public static void main(String[] args) {
        String texto = "cadena";
        int entero = 4;
        Clase instancia = new Clase(texto);
        metodo(entero, texto, instancia);
        System.out.println("Fuera de metodo:");
        System.out.println(entero);
        System.out.println(texto);
        System.out.println(instancia.a);
    }
}

```

Solución

En metodo:

5

cadena HOLA

cadena HOLA

Fuera de metodo:

4

cadena

cadena HOLA

Enunciado

¿Cómo puede arreglarse el siguiente programa?

```

import java.util.ArrayList;

abstract class A {
    int a;
    public A(int a) {
        this.a = a;
    }
}

class B extends A {
    public int opera() {
        return this.a + 1;
    }
}

class C extends A {
    public int opera() {
        return this.a + 2;
    }
}

```

```

public class Main {
    public static void main(String[] args) {
        ArrayList<A> miarray = new ArrayList<A>();
        miarray.add(new B(3));
        miarray.add(new C(5));
        for (A elemento : miarray) {
            System.out.println(elemento.opera());
        }
    }
}

```

Solución

El programa tiene varios problemas:

- En primer lugar, el método `opera()`, aunque está definido en ambas subclases, no está disponible para elementos de tipo A. Debe definirse de manera abstracta en A.
- En segundo lugar, las clases B y C tienen que tener un constructor.

El programa quedaría así:

```

import java.util.ArrayList;

abstract class A {
    int a;
    public A(int a) {
        this.a = a;
    }
    abstract public int opera();
}

class B extends A {
    public B(int a) {
        super(a);
    }

    public int opera() {
        return this.a + 1;
    }
}

class C extends A {
    public C(int a) {
        super(a);
    }

    public int opera() {
        return this.a + 2;
    }
}

```

```

public class Main {
    public static void main(String[] args) {
        ArrayList<A> miarray = new ArrayList<A>();
        miarray.add(new B(3));
        miarray.add(new C(5));
        for (A elemento : miarray) {
            System.out.println(elemento.opera());
        }
    }
}

```

Enunciado

¿Qué resultado imprime en pantalla el siguiente programa?

```

class A {
    A otro;
    public A(A otro) {
        this.otro = otro;
        System.out.println("En constructor de A con tipo A");
    }

    public A(B otro) {
        this.otro = otro;
        System.out.println("En constructor de A con tipo B");
    }

    public static void metodoEstatico() {
        System.out.println("En metodo estatico de A");
    }

    public void metodo() {
        System.out.println("En metodo de A");
    }
}

class B extends A {
    public B(A otro) {
        super(otro);
        System.out.println("En constructor de B con tipo A");
    }

    public B(B otro) {
        super((A) otro);
        System.out.println("En constructor de B con tipo B");
    }

    public static void metodoEstatico() {
        System.out.println("En metodo estatico de B");
    }
}

```

```
        public void metodo() {
            System.out.println("En metodo de B");
        }
    }

    public class Main {
        public static void main(String[] args) {
            B b = new B((B) null);
            A a = new A(b);
            A a2 = new A((A) b);
            ((A) b).metodo();
            ((A) b).metodoEstatico();
        }
    }
}
```

Solución

El programa imprime las siguientes líneas:

En constructor de A con tipo A
En constructor de B con tipo B
En constructor de A con tipo B
En constructor de A con tipo A
En metodo de B
En metodo estatico de A