

1 considere el siguiente programa (5 Puntos)

```
interface GFG {
    void myMethod();
    void getInfo();
}

abstract class Geeks implements GFG
{
    void getData(){
        System.out.println("GFG");
    }
}

public class Test extends Geeks
{
    public void myMethod(String message){
        System.out.println(message);
    }
    public void getInfo(){
        System.out.println("Geeks");
    }
    public static void main(String[] args){
        Geeks obj = new Test();
        obj.myMethod("GeeksforGeeks");
    }
}
```

¿Cuál sería la salida?

Error de ejecución
GeeksforGeeks
Error de compilación
Ninguna de las anteriores

Justifique:

2. ANALICE EL CÓDIGO QUE SE MUESTRA A CONTINUACIÓN

```
public interface Editable {
    int obtenerNumero();
    default String obtenerNombre() {
        return "Nombre por defecto";
    }
    static int obtenerCodigoPais() {
        return 0;
    }
}

class Contactotelefonico implements Editable {
    public int obtenerNumero() {
        return 987354723;
    }
    public String obtenerNombre() {
        return "Carlos Marín";
    }
    static int obtenerCodigoPais() {
        return 593;
    }
}

class AgendaTelefonica {
    public static void main(String args[]) {
        Editable obj = new Contactotelefonico();
    }
}
```

```
//sentencias de la tabla
}
}
```

Indique que retornan cada una de las sentencias a continuación, que son ejecutadas en el main de la clase AgendaTelefonica. (Justifique su respuesta brevemente)

	sentencia	Valor que retorna
a)	obj.obtenerNombre()	
b)	Editable.obtenerCodigoPais()	
c)	obj.obtenerNumero()	

d) Se desea definir **una nueva clase NO ABSTRACTA** que implemente la interfaz **ContactoTelefonicoI** ¿qué métodos debería implementar la clase?

3 - Interfaces

El siguiente código contiene tres errores, identifique los errores y como los corregirías.

```
1. interface A {
2.     void m1();
3.     String m2(){
4.         System.out.println("hola");
5.     }
6.     void m3();
7. }
8. public class B implements A {
9.     void m1(){
10.         System.out.println("One");
11.     }
12. }
```