

Apartado	Máx	Puntos	Comentarios
Clase CuentaBancaria	0,50	0	La clase Persona no debería ser abstracta. Los métodos de Imprimible deberían implementarse en esta clase. Además, la relación entre Persona y CuentaBancaria es de composición, no de herencia. La persona es un atributo de la cuenta (una cuenta no hereda de persona), incluso podría haber más de un titular para una misma cuenta. El método setSaldo no es muy apropiado para manejar el saldo. No se han implementado métodos ingresar-retirar que deberían estar en la clase CuentaCorriente.
Clase CuentaAhorro	0,50	0,4	No se implementan los métodos de Imprimible.
Colección lista de entidades autorizadas en CuentaCorriente	0,50	0,50	Bien la creación de la clase Entidades, aunque esta clase debería ocultar (encapsular) la estructura donde se almacenan los datos.
Clase CuentaCorrientePersonal	0,50	0,4	No se implementan los métodos de Imprimible.
Clase CuentaCorrienteEmpresa	0,50	0,25	No se implementan los métodos de Imprimible. Aquí se debería redefinir el método retirar ya que este tipo de cuenta permite que se quede al descubierto.
Colección para la lista de cuentas	1	0,25	Solo es necesaria una estructura de datos que almacene toda la lista de cuentas bancarias: ArrayList<CuentaBancaria>. Luego es posible recuperar o saber qué subtipo de CuentaBancaria es con el operador instanceof. Es mejor mantener la estructura con la lista de cuentas en memoria para que el acceso sea más rápido y guardar en disco cuando se cree una nueva, por ejemplo.
Control de errores y validación de datos	1	0,5	No se comprueban los datos y parámetros en las clases. La comprobación si está bien realizada en las ventanas del programa principal.
Uso de polimorfismo y ligadura dinámica	1	1	Bien.
Documentación de las clases	1	1	Bien.
Informe explicativo	1	1	Bien.
Programa principal	2,5	2,5	Bien.
<b>Total</b>	<b>10</b>	<b>7,8</b>	

