



TABLA 8: CE y Cb  
 Familia Profesional: Informática y Comunicaciones  
 Ciclo Formativo: Sistemas Microinformáticos y en Red  
 Módulo Profesional: Fundamentos de Programación

RA1	Reconoce la estructura de un programa informático, identificando y relacionando los elementos propios del lenguaje de programación utilizado.	1.- Fundamentos de la programación estructurada.		Saber Hacer	Saber Estar
Criterios de Evaluación	a) Se han identificado los bloques que componen la estructura de un programa informático.	<ul style="list-style-type: none"> <li>Estructura y bloques fundamentales.</li> <li>Comentarios.</li> </ul>	Contenidos Básicos	<ul style="list-style-type: none"> <li>Analizar programas para reconocer sus bloques fundamentales y comentarios.</li> </ul>	<ul style="list-style-type: none"> <li>Mostrar atención al detalle y disposición para comprender la estructura lógica.</li> </ul>
	b) Se han identificado los distintos tipos de variables y la utilidad específica de cada uno.	<ul style="list-style-type: none"> <li>Variables.</li> <li>Tipos de datos               <ul style="list-style-type: none"> <li>Números (enteros y flotantes).</li> <li>Cadenas de texto (strings).</li> <li>Listas, tuplas y diccionarios.</li> </ul> </li> </ul>		<ul style="list-style-type: none"> <li>Clasificar variables según su tipo y aplicarlas en ejemplos.</li> </ul>	<ul style="list-style-type: none"> <li>Ser meticuloso en la elección del tipo de dato adecuado.</li> </ul>
	c) Se ha modificado el código de un programa para crear y utilizar variables.			<ul style="list-style-type: none"> <li>Editar código existente para declarar y utilizar variables correctamente.</li> </ul>	<ul style="list-style-type: none"> <li>Actuar con responsabilidad en la modificación del código.</li> </ul>
	d) Se han creado y utilizado constantes y literales.	<ul style="list-style-type: none"> <li>Literales.</li> <li>Constantes.</li> </ul>		<ul style="list-style-type: none"> <li>Definir constantes y literales en programas sencillos.</li> </ul>	<ul style="list-style-type: none"> <li>Mantener rigor en la aplicación de buenas prácticas.</li> </ul>
	e) Se han clasificado, reconocido y utilizado en expresiones los operadores del lenguaje.	<ul style="list-style-type: none"> <li>Operadores y expresiones.</li> <li>Clasificación de operadores:               <ul style="list-style-type: none"> <li>Aritméticos.</li> <li>Relacionales.</li> <li>Lógicos.</li> </ul> </li> </ul>		<ul style="list-style-type: none"> <li>Implementar expresiones con operadores.</li> </ul>	<ul style="list-style-type: none"> <li>Mostrar precisión y orden en la construcción de expresiones.</li> </ul>
	f) Se ha comprobado el funcionamiento de las conversiones de tipo explícitas e implícitas.	<ul style="list-style-type: none"> <li>Conversiones de tipo.</li> </ul>		<ul style="list-style-type: none"> <li>Realizar conversiones explícitas e implícitas en ejemplos prácticos.</li> </ul>	<ul style="list-style-type: none"> <li>Ser cuidadoso en la gestión de tipos para evitar errores.</li> </ul>



TABLA 8: CE y Cb  
 Familia Profesional: Informática y Comunicaciones  
 Ciclo Formativo: Sistemas Microinformáticos y en Red  
 Módulo Profesional: Fundamentos de Programación

RA1	Reconoce la estructura de un programa informático, identificando y relacionando los elementos propios del lenguaje de programación utilizado.	1.- Fundamentos de la programación estructurada.	Saber Hacer	Saber Estar
	h) Se ha utilizado la consola para realizar operaciones de entrada y salida de información.	<ul style="list-style-type: none"> <li>Entrada y salida de datos:               <ul style="list-style-type: none"> <li>- Capturar datos del usuario.</li> <li>- Mostrar resultados.</li> </ul> </li> </ul>	<ul style="list-style-type: none"> <li>Capturar datos del usuario y mostrar resultados en la consola.</li> </ul>	<ul style="list-style-type: none"> <li>Mostrar actitud proactiva y clara en la interacción con el usuario.</li> </ul>
	i) Se han reconocido las posibilidades de entrada / salida del lenguaje y las librerías asociadas.	<ul style="list-style-type: none"> <li>Ficheros               <ul style="list-style-type: none"> <li>- Apertura y cierre de ficheros.</li> <li>- Modos de apertura de ficheros (lectura y/o escritura).</li> </ul> </li> <li>Desplazamiento en ficheros.</li> </ul>	<ul style="list-style-type: none"> <li>Implementar lectura y escritura en ficheros, gestionando apertura y cierre.</li> </ul>	<ul style="list-style-type: none"> <li>Mostrar responsabilidad en la manipulación de datos.</li> </ul>

RA1	Reconoce la estructura de un programa informático, identificando y relacionando los elementos propios del lenguaje de programación utilizado.	2.- Fundamentos de la programación modular.	Saber Hacer	Saber Estar
	g) Se han identificado los ámbitos de utilización de las variables.	<ul style="list-style-type: none"> <li>Definición y creación de funciones.</li> <li>Parámetros y argumentos:               <ul style="list-style-type: none"> <li>- Definición.</li> <li>- Paso por valor.</li> <li>- Paso por referencia.</li> <li>- Valores de retorno (`return`).</li> </ul> </li> <li>Ámbito de las variables:</li> </ul>	<ul style="list-style-type: none"> <li>Definir funciones con parámetros y valores de retorno, aplicando variables locales y globales.</li> </ul>	<ul style="list-style-type: none"> <li>Ser riguroso en la organización del código.</li> </ul>



TABLA 8: CE y Cb  
 Familia Profesional: Informática y Comunicaciones  
 Ciclo Formativo: Sistemas Microinformáticos y en Red  
 Módulo Profesional: Fundamentos de Programación

RA1	Reconoce la estructura de un programa informático, identificando y relacionando los elementos propios del lenguaje de programación utilizado.	2.- Fundamentos de la programación modulara.	Saber Hacer	Saber Estar
		<ul style="list-style-type: none"> <li>- Variables locales.</li> <li>- Variables globales.</li> <li>• Uso de funciones del lenguaje.</li> <li>• Documentación de funciones.</li> <li>• Introducción a las pruebas y verificación del funcionamiento de funciones.</li> </ul>		

RA2	Escribe y depura código, analizando y utilizando las estructuras de control del lenguaje.	1.- fundamentos de la programación estructurada	Saber Hacer	Saber Estar
Criterios de Evaluación	a) Se han creado bucles y se ha verificado su funcionamiento.	<ul style="list-style-type: none"> <li>• Bucles for y while</li> </ul>	Contenidos Básicos	<ul style="list-style-type: none"> <li>• Actuar con rigor en la verificación del funcionamiento.</li> </ul>
	b) Se han utilizado mecanismos de decisión en la creación de bloques de sentencias.	<ul style="list-style-type: none"> <li>• Sentencias condicionales</li> </ul>		<ul style="list-style-type: none"> <li>• Mantener una actitud analítica y ordenada.</li> </ul>
	c) Identifica los distintos tipos de sentencias de decisión existentes.	<ul style="list-style-type: none"> <li>• Estructuras de selección.               <ul style="list-style-type: none"> <li>- `if`</li> <li>- `else`</li> <li>- `elif`.</li> </ul> </li> </ul>		<ul style="list-style-type: none"> <li>• Mostrar disposición para aprender y aplicar buenas prácticas.</li> </ul>
	d) Diferencia los distintos tipos de bucles repetitivos.	<ul style="list-style-type: none"> <li>• Estructuras de repetición.               <ul style="list-style-type: none"> <li>- - `for`</li> <li>- - `while`).</li> </ul> </li> <li>• Estructuras de salto.</li> </ul>		<ul style="list-style-type: none"> <li>• Ser meticuloso en la elección de estructuras.</li> </ul>



TABLA 8: CE y Cb  
 Familia Profesional: Informática y Comunicaciones  
 Ciclo Formativo: Sistemas Microinformáticos y en Red  
 Módulo Profesional: Fundamentos de Programación

RA2	Escribe y depura código, analizando y utilizando las estructuras de control del lenguaje.	1.- fundamentos de la programación estructurada	Saber Hacer	Saber Estar
		<ul style="list-style-type: none"> <li>- Continue.</li> <li>- Break.</li> <li>- Pass.</li> </ul>		
	e) Se han creado condiciones sencillas de decisión.		<ul style="list-style-type: none"> <li>• Formular condiciones simples en sentencias condicionales.</li> </ul>	<ul style="list-style-type: none"> <li>• Demostrar claridad y precisión en la expresión de condiciones.</li> </ul>
	f) Se han creado sentencias de control complejas.		<ul style="list-style-type: none"> <li>• Combinar estructuras condicionales y bucles para resolver problemas elaborados.</li> </ul>	<ul style="list-style-type: none"> <li>• Mostrar iniciativa y creatividad en la resolución de problemas.</li> </ul>
	g) Se ha comentado y documentado el código.		<ul style="list-style-type: none"> <li>• Incluir comentarios explicativos y documentación básica.</li> </ul>	<ul style="list-style-type: none"> <li>• Ser responsable y ordenado en la presentación del trabajo.</li> </ul>

RA3	Escribe y prueba programas sencillos, reconociendo y aplicando los fundamentos de la programación orientada a objetos.	3.- Fundamentos de programación orientada a objetos.		Saber Hacer	Saber Estar
Criterios de Evaluación	a) Se han identificado los fundamentos de la programación orientada a objetos.	<ul style="list-style-type: none"><li>• Concepto de clase.</li><li>• Estructura y miembros de una clase.</li><li>• Visibilidad.</li><li>• Creación de propiedades.</li></ul>	Contenidos Básicos	<ul style="list-style-type: none"><li>• Analizar ejemplos para reconocer clases, objetos, atributos y métodos.</li></ul>	<ul style="list-style-type: none"><li>• Mostrar interés por comprender la estructura y principios de la POO.</li></ul>
	b) Se han escrito programas simples.	<ul style="list-style-type: none"><li>• Programas con clases y objetos</li></ul>		<ul style="list-style-type: none"><li>• Crear programas básicos que implementen clases y objetos.</li></ul>	<ul style="list-style-type: none"><li>• Actuar con responsabilidad en la elaboración del código.</li></ul>
	c) Se han instanciado objetos a partir de clases predefinidas.	<ul style="list-style-type: none"><li>• Instanciación de objetos</li></ul>		<ul style="list-style-type: none"><li>• Utilizar clases existentes para generar objetos funcionales.</li></ul>	<ul style="list-style-type: none"><li>• Ser ordenado y preciso en la instanciación y uso de objetos.</li></ul>



TABLA 8: CE y Cb  
 Familia Profesional: Informática y Comunicaciones  
 Ciclo Formativo: Sistemas Microinformáticos y en Red  
 Módulo Profesional: Fundamentos de Programación

RA3	Escribe y prueba programas sencillos, reconociendo y aplicando los fundamentos de la programación orientada a objetos.	3.- Fundamentos de programación orientada a objetos.	Saber Hacer	Saber Estar
	d) Se han utilizado métodos y propiedades de los objetos.	<ul style="list-style-type: none"> <li>Métodos getters y setters.</li> </ul>	<ul style="list-style-type: none"> <li>Implementar métodos getters y setters para gestionar atributos.</li> </ul>	<ul style="list-style-type: none"> <li>Mantener una actitud meticulosa en la manipulación de datos.</li> </ul>
	e) Se han utilizado parámetros en la llamada a métodos.	<ul style="list-style-type: none"> <li>Parámetros.</li> <li>Utilización de propiedades.</li> <li>Utilización de métodos estáticos.</li> <li>Constructores.</li> <li>Destrucción de objetos y liberación de memoria.</li> </ul>	<ul style="list-style-type: none"> <li>Definir y pasar parámetros en métodos, aplicando constructores.</li> </ul>	<ul style="list-style-type: none"> <li>Mostrar rigor en la definición y uso de parámetros.</li> </ul>
	f) Se han incorporado y utilizado librerías de objetos.	<ul style="list-style-type: none"> <li>Librerías externas</li> </ul>	<ul style="list-style-type: none"> <li>Integrar librerías externas para ampliar funcionalidades.</li> </ul>	<ul style="list-style-type: none"> <li>Ser proactivo en la búsqueda de soluciones y recursos adecuados.</li> </ul>
	g) Se han utilizado constructores.	<ul style="list-style-type: none"> <li>Encapsulamiento:               <ul style="list-style-type: none"> <li>Uso de modificadores de acceso (público, privado).</li> </ul> </li> </ul>	<ul style="list-style-type: none"> <li>Implementar constructores para inicializar objetos correctamente.</li> </ul>	<ul style="list-style-type: none"> <li>Mostrar compromiso con la calidad del código y su mantenimiento.</li> </ul>
	h) Se han creado excepciones.	<ul style="list-style-type: none"> <li>Excepciones y manejo de errores (uso de `try`, `except`).</li> </ul>	<ul style="list-style-type: none"> <li>Aplicar bloques try y except para manejar errores.</li> </ul>	<ul style="list-style-type: none"> <li>Actuar con responsabilidad en la prevención y gestión de errores.</li> </ul>



TABLA 8: CE y Cb  
 Familia Profesional: Informática y Comunicaciones  
 Ciclo Formativo: Sistemas Microinformáticos y en Red  
 Módulo Profesional: Fundamentos de Programación

RA4	Desarrolla programas organizados en clases analizando y aplicando los principios de la programación orientada a objetos	3.- Fundamentos de programación orientada a objetos	Saber Hacer	Saber Estar
Criterios de Evaluación	a) Se han desarrollado programas que instancien y utilicen objetos de las clases creadas anteriormente.		<ul style="list-style-type: none"> <li>Diseñar programas completos que combinen varias clases y objetos.</li> </ul>	<ul style="list-style-type: none"> <li>Mostrar constancia y organización en el desarrollo del programa.</li> </ul>
	b) Se han utilizado mecanismos para controlar la visibilidad de las clases y de sus miembros.		<ul style="list-style-type: none"> <li>Aplicar modificadores de acceso para garantizar encapsulación.</li> </ul>	<ul style="list-style-type: none"> <li>Ser cuidadoso en la protección de datos y la estructura del código.</li> </ul>
	c) Se han definido y utilizado clases heredadas.	<ul style="list-style-type: none"> <li>Composición de clases.</li> <li>Herencia y polimorfismo.</li> <li>Jerarquía de clases:               <ul style="list-style-type: none"> <li>Superclases.</li> <li>Subclases.</li> </ul> </li> <li>Clases y métodos abstractos y finales.</li> <li>Interfaces.</li> <li>Sobreescritura de métodos.</li> <li>Polimorfismo:               <ul style="list-style-type: none"> <li>Uso de métodos sobrescritos en clases derivadas.</li> </ul> </li> <li>Métodos especiales:               <ul style="list-style-type: none"> <li>Uso de métodos mágicos como <code>`__init__()`</code>, <code>`__str__()`</code>, y <code>`__repr__()`</code>.</li> </ul> </li> <li>Constructores y herencia.</li> </ul>	<ul style="list-style-type: none"> <li>Implementar herencia y polimorfismo para reutilizar código.</li> </ul>	<ul style="list-style-type: none"> <li>Mostrar actitud crítica y reflexiva en la elección de estructuras.</li> </ul>
	d) Se han creado y utilizado métodos estáticos.		<ul style="list-style-type: none"> <li>Definir métodos estáticos para operaciones que no dependen de instancias.</li> </ul>	<ul style="list-style-type: none"> <li>Ser preciso y coherente en la aplicación de principios de diseño.</li> </ul>



# TABLA 8: CE y Cb

Familia Profesional: Informática y Comunicaciones  
Ciclo Formativo: Sistemas Microinformáticos y en Red  
Módulo Profesional: Fundamentos de Programación