



**CICLO FORMATIVO DE
GRADO SUPERIOR
DESARROLLO DE APLICACIONES WEB
MÓDULO:
PROGRAMACIÓN
MODALIDAD DE DISTANCIA**

Programación didáctica.

Módulo 0485
PROGRAMACIÓN. (PROGR)

Profesor

CARLOS BLANCO MEDIO

Identificación de la programación

Centro Educativo

Código	Centro	Año académico
33028167	CIFP Sectores Industriales y Servicios	2019 /2020

Ciclo formativo

Familia profesional	Ciclo formativo	Código	Curso
Informática y Comunicaciones	Desarrollo de Aplicaciones Web. (Distancia)	DIFC303	1º

LEGISLACIÓN APLICABLE	IFC303 Desarrollo de Aplicaciones Web Real Decreto 686/2010 del Ministerio de Educación, de 20 de mayo, por el que se establece el título de Técnico Superior en Desarrollo de Aplicaciones Web y se fijan sus enseñanzas mínimas. Decreto 184/2012 de la Consejería de Educación, Cultura y Deporte del Principado de Asturias, de 8 de agosto, por el que se establece el currículo del ciclo formativo de Grado Superior de Formación Profesional en Desarrollo de Aplicaciones Web.
Nº de créditos	ECTS: 14
Horas/curso	224 horas
Horas/semana	4 horas, distribuidas en 3 horas para tutoría colectiva y 1 hora para tutoría individual
Especialidad del profesorado	PROFESOR DE ENSEÑANZA SECUNDARIA – 107 - Informática

1. RESULTADOS DE APRENDIZAJE QUE DEFINEN EL MÓDULO

1.- Reconoce la estructura de un programa informático, identificando y relacionando los elementos propios del lenguaje de programación utilizado.
2.- Escribe y prueba programas sencillos, reconociendo y aplicando los fundamentos de la programación orientada a objetos.
3.- Escribe y depura código, analizando y utilizando las estructuras de control del lenguaje.
4.- Desarrolla programas organizados en clases analizando y aplicando los principios de la programación orientada a objetos.
5.- Realiza operaciones de entrada y salida de información, utilizando procedimientos específicos del lenguaje y librerías de clases.
6.- Escribe programas que manipulen información, seleccionando y utilizando tipos avanzados de datos.
7.- Desarrolla programas, aplicando características avanzadas de los lenguajes orientados a objetos y del entorno de programación.
8.- Desarrolla programas, aplicando características específicas de los lenguajes orientados a objetos y con programación estructurada.
9.- Gestiona información almacenada en bases de datos relacionales manteniendo la integridad y consistencia de los datos.

2. CRITERIOS DE EVALUACIÓN MATIZADOS Y RELACIÓN CON LOS RESULTADOS DE APRENDIZAJE

Resultados de Aprendizaje	Criterios de Evaluación
1.- Reconoce la estructura de un programa informático, identificando y relacionando los elementos propios del lenguaje de programación utilizado.	a) Se han identificado los bloques que componen la estructura de un programa informático. b) Se han creado proyectos de desarrollo de aplicaciones c) Se han utilizado entornos integrados de desarrollo. d) Se han identificado los distintos tipos de variables y la utilidad específica de cada uno. e) Se ha modificado el código de un programa para crear y utilizar variables. f) Se han creado y utilizado constantes y literales. g) Se han clasificado, reconocido y utilizado en expresiones los operadores del lenguaje. h) Se ha comprobado el funcionamiento de las conversiones de tipo explícitas e implícitas. i) Se han introducido comentarios en el código.
2.- Escribe y prueba programas sencillos, reconociendo y aplicando los fundamentos de la programación orientada a objetos.	a) Se han identificado los fundamentos de la programación orientada a objetos. b) Se han escrito programas simples. c) Se han instanciado objetos a partir de clases predefinidas. d) Se han utilizado métodos y propiedades de los objetos. e) Se han escrito llamadas a métodos estáticos. f) Se han utilizado parámetros en la llamada a métodos. g) Se han incorporado y utilizado librerías de objetos.

	<ul style="list-style-type: none"> h) Se han utilizado constructores. i) Se ha utilizado el entorno integrado de desarrollo en la creación y compilación de programas simples.
3.- Escribe y depura código, analizando y utilizando las estructuras de control del lenguaje.	<ul style="list-style-type: none"> a) Se ha escrito y probado código que haga uso de estructuras de selección. b) Se han utilizado estructuras de repetición. c) Se han reconocido las posibilidades de las sentencias de salto. d) Se ha escrito código utilizando control de excepciones. e) Se han creado programas ejecutables utilizando diferentes estructuras de control. f) Se han probado y depurado los programas. g) Se ha comentado y documentado el código.
4.- Desarrolla programas organizados en clases analizando y aplicando los principios de la programación orientada a objetos.	<ul style="list-style-type: none"> a) Se ha reconocido la sintaxis, estructura y componentes típicos de una clase. b) Se han definido clases. c) Se han definido propiedades y métodos. d) Se han utilizado constructores y destructores. e) Se han desarrollado programas que instancien y utilicen objetos de las clases creadas anteriormente. f) Se han utilizado mecanismos para controlar la visibilidad de las clases y de sus miembros. g) Se han definido y utilizado clases heredadas. h) Se han creado y utilizado métodos estáticos. i) Se han definido y utilizado interfaces. j) Se han creado y utilizado conjuntos y librerías de clases.
5.- Realiza operaciones de entrada y salida de información, utilizando procedimientos específicos del lenguaje y librerías de clases.	<ul style="list-style-type: none"> a) Se ha utilizado la consola para realizar operaciones de entrada y salida de información. b) Se han aplicado formatos en la visualización de la información. c) Se han reconocido las posibilidades de entrada / salida del lenguaje y las librerías asociadas. d) Se han utilizado ficheros para almacenar y recuperar información. e) Se han creado programas que utilicen diversos métodos de acceso al contenido de los ficheros. f) Se han implementado técnicas para envío y almacenamiento de objetos en ficheros. g) Se han utilizado las herramientas del entorno de desarrollo para crear interfaces gráficas de usuario simples. h) Se han programado controladores de eventos. i) Se han escrito programas que utilicen interfaces gráficas para la entrada y salida de información.
6.- Escribe programas que manipulen información, seleccionando y utilizando tipos avanzados de datos.	<ul style="list-style-type: none"> a) Se han escrito programas que utilicen arrays. b) Se han reconocido las librerías de clases relacionadas con tipos de datos avanzados. c) Se han utilizado listas para almacenar y procesar información. d) Se han utilizado iteradores para recorrer los elementos de las listas. e) Se han reconocido las características y ventajas de cada una de la colecciones de datos disponibles. f) Se han creado clases y métodos genéricos. g) Se han utilizado expresiones regulares en la búsqueda de patrones en cadenas de texto.

7.- Desarrolla programas, aplicando características avanzadas de los lenguajes orientados a objetos y del entorno de programación.	<ul style="list-style-type: none"> a) Se han identificado los conceptos de herencia, superclase y subclase. b) Se han utilizado modificadores para bloquear y forzar la herencia de clases y métodos. c) Se ha reconocido la incidencia de los constructores en la herencia. d) Se han creado clases heredadas que sobrescriban la implementación de métodos de la superclase. e) Se han diseñado y aplicado jerarquías de clases. f) Se han probado y depurado las jerarquías de clases. g) Se han realizado programas que implementen y utilicen jerarquías de clases. h) Se ha comentado y documentado el código.
8.- Desarrolla programas, aplicando características específicas de los lenguajes orientados a objetos y con programación estructurada.	<ul style="list-style-type: none"> a) Se han identificado las nuevas estructuras de datos en la programación estructuradas. b) Se ha analizado su aplicación en el desarrollo de aplicaciones. c) Se han utilizado las nuevas estructuras en el desarrollo de aplicaciones específicas en el ámbito de la programación estructurada. d) Se han utilizado las nuevas estructuras en el desarrollo de aplicaciones específicas en el ámbito de la programación orientada a objetos.
9.- Gestiona información almacenada en bases de datos relacionales manteniendo la integridad y consistencia de los datos.	<ul style="list-style-type: none"> a) Se han identificado las características y métodos de acceso a sistemas gestores de bases de datos relacionales. b) Se han programado conexiones con bases de datos. c) Se ha escrito código para almacenar información en bases de datos. d) Se han creado programas para recuperar y mostrar información almacenada en bases de datos. e) Se han efectuado borrados y modificaciones sobre la información almacenada. f) Se han creado aplicaciones que ejecuten consultas sobre bases de datos. g) Se han creado aplicaciones para posibilitar la gestión de información presente en bases de datos relacionales.

3. CONTENIDOS DE LAS UNIDADES DE TRABAJO

UNIDAD DE TRABAJO	1 Elementos de un programa informático
CONCEPTOS	<ul style="list-style-type: none"> 1.- Conceptos iniciales. 2.- Sistemas de procesamiento de la información. Algoritmo. 3.- Aplicación informática. Ciclo de vida. 4.- Lenguajes de programación. 5.- Estructuras Básicas de datos. Identificadores. Datos. Tipos. Expresiones. Funciones. Operadores. Prioridad. 6.- Herramientas de programación. Pseudocódigo. 7.- Partes principales de un programa. 8.- Clasificación de las instrucciones. 9.- Programación Modular. Funciones

UNIDAD DE TRABAJO	2 Lenguaje de programación java: primeros pasos
CONCEPTOS	1.- Orígenes del lenguaje Java: Un poco de historia 2.- Características de Java. 3.- Conceptos básicos sobre Java. 4.- Literales en Java. Boolean. Char. Enteros. Reales. 5.- Identificadores en Java. 6.- Operadores en Java 7.- Tipos de sentencias en Java. 8.- Sentencias de expresión. 9.- Sentencias de declaración. 10.- Sentencias de control de flujo. 11. Entrada y salida básica 12. Funciones o métodos. Recursividad

UNIDAD DE TRABAJO	3 Fundamentos de programación orientada a objetos
CONCEPTOS	1.- Programación estructurada y POO. 2.- Conceptos iniciales 3.- Conceptos fundamentales de la POO 4.- Las Clases

UNIDAD DE TRABAJO	4 Clases y objetos en java
CONCEPTOS	<ul style="list-style-type: none"> - Definición de clases en Java. - Control de acceso en Java. - Objetos o instancias de una clase. - Miembros de objeto y miembros de clase. - Referencias a Objetos. - Métodos o mensajes. - Constructores.

UNIDAD DE TRABAJO	5 Utilización avanzada de clases en java
CONCEPTOS	<ul style="list-style-type: none"> - Acceso a atributos privados. - Métodos getter y setter. - Sobrecarga de métodos. - Constructores sobrecargados. - Operador this. - Herencia. Control de acceso. Redefinición de métodos. - Polimorfismo. - Clases Abstractas.

UNIDAD DE TRABAJO	6 Estructuras de almacenamiento en java
CONCEPTOS	<ul style="list-style-type: none"> - Introducción - Tablas, matrices, vectores o arrays - Declaración y dimensionamiento de arrays - Recorrido y llenado de arrays - Procesamiento de un array. Parámetros. - Ordenación y búsqueda de elementos. - Ordenación de arrays. - Manejo de excepciones. Creación de excepciones. - Cadenas de caracteres. Clase String.

UNIDAD DE TRABAJO	7 Colecciones de datos en java
CONCEPTOS	<ul style="list-style-type: none"> - Tipos de colecciones. - Jerarquía de colecciones. - Interfaz Collection - Operaciones con colecciones: Acceso a los elementos y recorridos - Uso de las clases más importantes y significativas y sus métodos principales: ArrayList, LinkedList, HashSet.

UNIDAD DE TRABAJO	8 Lectura y escritura de información en java
CONCEPTOS	<ul style="list-style-type: none"> - Introducción. Flujos de datos. Clase File. - Clases relativas a flujos de datos. - Flujos de Bytes. - Flujos de caracteres. - Lectura y escritura de datos binarios. - Archivos de acceso aleatorio. - Almacenamiento de objetos en ficheros: Persistencia. Serialización.

UNIDAD DE TRABAJO	9 Gestión de bases de datos relacionales
CONCEPTOS	<ul style="list-style-type: none"> - Interfaces de programación de acceso a las bases de datos - Establecimiento de conexiones. - Recuperación de la información contenida en las B.D. - Manipulación de la información contenida en las B.D.: creación de tablas, inserción, actualización y eliminación de registros

RESULTADOS DE APRENDIZAJE									UNIDAD DE TRABAJO		
R1	R2	R3	R4	R5	R6	R7	R8	R9			
X									1	Elementos de un programa informático.	1º EVALUACIÓN
		X							2	Lenguaje de programación Java: Primeros pasos.	
	X						X		3	Fundamentos de la Programación Orientada a Objetos (POO).	
	X		X						4	Clases y Objetos en Java.	2º EVALUACIÓN
			X			X			5	Utilización avanzada de Clases en Java.	
		X			X				6	Estructuras de almacenamiento en Java.	
					X				7	Colecciones de datos en Java.	3º EVALUACIÓN
				X					8	Lectura y escritura de información en Java.	
								X	9	Gestión de bases de datos relacionales.	

4. PROCEDIMIENTO E INSTRUMENTOS DE EVALUACIÓN

Los instrumentos de evaluación serán los siguientes:

1. Exámenes a lo largo de la evaluación.- Serán tres exámenes, uno antes de cada evaluación e incluirán los contenidos de todos los temas vistos en el trimestre. Tendrán como misión la evaluación del grado de conocimiento de los contenidos y del trabajo realizado. Se calificarán sobre un total de 10 puntos.
En el encabezado de los exámenes constará una nota sobre la valoración de los ejercicios, que será la siguiente: <<Se valorarán positivamente las siguientes cuestiones: Correcta identificación en los programas, comentarios y correcta autodocumentación de los mismos y claridad, versatilidad y eficiencia del algoritmo, así como el uso de excepciones en aquellas partes del código que así lo requieran o que se consideren necesarias.>>
2. Las tareas evaluables desarrollados individualmente. Su finalidad será evaluar la realización de las prácticas de los alumnos, su documentación y el funcionamiento adecuado de los programas propuestos. La entrega y supervisión de estos trabajos será llevada a cabo a través de la entrega de tareas por el aula virtual. Habrá un máximo de dos tareas evaluables y un mínimo de una por cada trimestre.

5. CRITERIOS DE EVALUACIÓN

Evaluaciones trimestrales.

Evaluación mediante pruebas trimestrales, celebradas en las convocatorias que corresponden a cada trimestre, son el medio de control del rendimiento de los alumnos e instrumento primordial de calificación, dada la condición del alumnado de distancia. Las pruebas serán de carácter práctico (resolver supuestos). Tiene carácter VOLUNTARIO y se desarrollarán en las fechas establecidas al respecto y determinadas en el calendario que se proporciona al principio del curso.

Los exámenes consistirán en resolver una serie de problemas en pseudocódigo o en lenguaje java, que se responderán, en principio, sobre papel. Si hubiera ordenadores suficientes, se podrá hacer sobre ordenador aquel alumnado que así lo prefiera, utilizando el entorno de desarrollo escogido por el alumno, BlueJ o Eclipse. En el caso de hacer el examen en ordenador, las soluciones propuestas deberán ser “subidas” al aula virtual de este módulo, por lo que será necesario que este alumnado conozca su identidad digital de educastur y su clave para acceder a la misma.

En las sesiones previas a la evaluación se trabajará la tipología de la prueba prevista y se subirá al aula virtual información sobre la misma.

En cada se realizarán una o dos tareas evaluables. Deberán ser entregadas en plazo y forma según se acuerde, a través del aula virtual del módulo. Se valorará su calidad, el cumplimiento de objetivos, la completitud y la funcionalidad (si procede).

La nota de cada trimestre se obtendrá de la media ponderada de los siguientes aspectos:

Prueba objetiva del periodo: 80%

Tareas evaluables en el periodo: 20%

Para obtener una nota de aprobado en cada una de las evaluaciones, la calificación de la prueba objetiva (EXAMEN) no debe ser inferior a 4.

Evaluación final ordinaria.

La calificación del alumno al finalizar el curso será la media aritmética de las calificaciones obtenidas en cada trimestre. Para resolver el ajuste a la nota final entera entre 1 y 10, se tendrá en cuenta la mayor preponderancia de la segunda y tercera evaluaciones en la que se utilizan todos los conceptos adquiridos durante el curso.

Debido a que los contenidos de este módulo son agregativos y una vez conocidos continúan siendo utilizados continuamente a partir de su impartición, se considera que aprobando una de las evaluaciones se conocen y superan los contenidos de las anteriores.

Aquellos alumnos que no superen alguna de las evaluaciones trimestrales, tendrán una oportunidad de recuperación en la prueba global que tendrá lugar en el mes de junio. El profesor, a través del aula virtual indicará a cada alumno aquellas evaluaciones que tiene pendiente, teniendo en cuenta las características especiales de los contenidos agregativos de este módulo.

Evaluación final extraordinaria: EXAMEN DE SEPTIEMBRE

Alumnos a los que va dirigido:

Realizarán este examen aquellos alumnos que no han aprobado el módulo en la convocatoria de junio.

El tutor comunicará a través del aula virtual las posibles actividades de recuperación para que el alumnado suspenso prepare esta prueba, que servirán de guía para preparar el examen de septiembre.

Estructura de la prueba de Septiembre: Constará de un examen global que determine sus conocimientos sobre todos los contenidos del módulo. El examen será una prueba para comprobar que el alumno domina los mínimos del módulo, englobando la totalidad de las unidades temáticas en las que está dividida la programación.

Consistirá en realizar varios programas, sobre la base de todo el temario, en el lenguaje visto durante el curso para resolver determinados problemas, aplicando la metodología y las estructuras de datos oportunas. Se realizará por escrito o en el ordenador, con los entornos de programación usados durante el curso, al igual que en las pruebas de evaluación o global de junio.

Para superar esta prueba y obtener una calificación de 5 (cinco) en el módulo, el alumno debe demostrar que domina de manera suficiente los resultados de aprendizaje del módulo. La no superación de alguno de estos conllevará una nota de suspenso (menor que cinco) en esta convocatoria.