



GOBIERNO DE
MÉXICO

EDUCACIÓN
SECRETARÍA DE EDUCACIÓN PÚBLICA



Programa de estudios del módulo

Programación orientada a objetos

Currículum Laboral

Áreas:

Tecnología y transporte

Carreras:

Profesional Técnico-Bachiller en
Informática

4º semestre

Editor: Colegio Nacional de Educación Profesional Técnica

Módulo: Programación orientada a objetos.

Área: Tecnología y transporte.

Carrera: PT-B en Informática.

Semestre: Cuarto.

Horas por semana: 8

Fecha de diseño o actualización: 31 de mayo del 2024.

Vigencia: a partir de la aprobación de la Junta Directiva y en tanto no se genere un documento que lo actualice.

© Colegio Nacional de Educación Profesional Técnica

Prohibida la reproducción total o parcial de esta obra por cualquier medio, sin autorización por escrito del CONALEP.

Directorio

Arturo Pontifes Martínez

Dirección General

Camilo García Ramírez

Secretaría General

Hugo Nicolás Pérez González

Secretaría Académica

Patricia Alejandra Bernal Monzón

Dirección de Diseño Curricular

Programación orientada a objetos

Contenido		Pág.
Capítulo I:	Generalidades del Profesional Técnico-Bachiller	
1.1	Marco Curricular Común de la Educación Media Superior	5
1.2	Objetivo de la Carrera	6
Capítulo II:	Aspectos Específicos del Módulo	
2.1	Presentación	7
2.2	Propósito del Módulo	9
2.3	Mapa del Módulo	10
2.4	Unidades de Aprendizaje	11
2.5	Referencias	17

CAPÍTULO I: Generalidades del Profesional Técnico-Bachiller

1.1 Marco Curricular Común de la Educación Media Superior

El Marco Curricular Común de la Educación Media Superior propone una apuesta curricular centrada en el desarrollo integral de las y los adolescentes y jóvenes, con la finalidad de formar estudiantes capaces de conducir su vida hacia su futuro con bienestar y satisfacción; con sentido de pertenencia social, conscientes de los problemas sociales, económicos y políticos que aquejan al país, dispuestos a participar de manera responsable y con toma de decisión hacia los procesos de la democracia participativa y compromiso por generar soluciones de las problemáticas que los aquejan y que tengan la capacidad de aprender a aprender en el trayecto de su vida. Que sean adolescentes y jóvenes capaces de erigirse como agentes de transformación social y que fomenten una cultura de paz y de respeto hacia la diversidad social, sexual, política y étnica; solidarios y empáticos.

A través del currículum laboral, el Profesional Técnico-Bachiller desarrollará competencias básicas y extendidas pertinentes, buscando la transversalidad con los módulos del currículum fundamental y ampliado; permitiendo con ello desarrollar habilidades, conocimientos y actitudes para comprender los procesos productivos en los que está involucrado para enriquecerlos, transformarlos, resolver problemas, ejercer la toma de decisiones y desempeñarse en diferentes ambientes laborales, con una actitud creadora, crítica, responsable y propositiva; de la misma manera, fomenta el trabajo en equipo, colaborativo, el desarrollo pleno de su potencial en los ámbitos profesional, personal, así como la convivencia de manera armónica con el medio ambiente y la sociedad.

1.2 Objetivo de la Carrera

PT-B en Informática

Desempeñar funciones técnico-operativas inherentes al desarrollo e implantación de soluciones de tecnologías de información basados en la automatización, organización, codificación, recuperación de la información y optimización de recursos informáticos a fin de impulsar la competitividad, las buenas prácticas y toma de decisiones en organizaciones o empresas de cualquier ámbito.

CAPÍTULO II: Aspectos Específicos del Módulo

2.1 Presentación

El módulo de Programación orientada a objetos pertenece al currículum laboral y se imparte en el cuarto semestre de la carrera de Profesional Técnico Bachiller en Informática. Tiene como finalidad que el alumno desarrolle aplicaciones de software, acorde con requerimientos del usuario y utilizando la metodología de programación orientada a objetos para agilizar y resolver problemas complejos relacionados con el procesamiento de información.

El módulo se divide en dos unidades de aprendizaje. En la primera unidad se establece el manejo de componentes del lenguaje orientados a objetos, modelando y codificando programas de cómputo haciendo uso del paradigma orientado a objetos, el uso de sentencias de control, objetos y clases. En la segunda unidad se desarrollan aplicaciones orientadas a objetos a través de la interacción de los objetos y actores del sistema, asegurando la integridad de los datos al implementar medidas de seguridad, utilizando patrones de diseño.

La contribución del módulo al perfil de egreso de la carrera en la que está considerado, incluye el desarrollo de habilidades para el manejo de recursos informáticos con base en herramientas de aplicaciones avanzadas, para elaborar documentos de texto, a fin de cubrir las necesidades de digitalización de documentos escritos de los usuarios, elaborar presentaciones ejecutivas gráficas para atender las necesidades de exposiciones, proyecciones, digitalización y presentaciones escritas de los usuarios, así como la aplicación de hojas de cálculo, el uso y aprovechamiento del correo electrónico, buscadores de información y navegadores en internet cuya presencia y evolución constante los hacen herramientas tecnológicas indispensables en las empresas.

La tarea educativa en este módulo tendrá que diversificarse, a fin de que los docentes realicen funciones preceptoras, que consistirán en la guía y acompañamiento del alumnado durante su proceso de formación académica y personal y en la definición de estrategias de participación que permitan incorporar a su familia en un esquema de corresponsabilidad que coadyuve a su desarrollo integral; por tal motivo, deberá destinar tiempo dentro de cada unidad para brindar este apoyo a la labor educativa de acuerdo con el Programa de Preceptorías.

Por otro lado, el alumnado deberá gestionar su aprendizaje, a fin de distribuir su tiempo para dedicar un porcentaje de la duración del módulo al estudio independiente, para reforzar el conocimiento previo o adquirido en clase, de tal forma que obtengan hábitos de estudio que le permitan ser autodidacta.

Finalmente, es necesario que al concluir cada resultado de aprendizaje se considere una sesión de clase en la cual se realice la recapitulación de los aprendizajes logrados, con el propósito de verificar que éstos se han alcanzado o en caso contrario, determinar las acciones de mejora pertinentes. Cabe señalar que en esta sesión el alumno o la alumna que haya obtenido insuficiencia en sus actividades de evaluación o desee mejorar su resultado, tendrá la oportunidad de entregar nuevas evidencias.

2.2 Propósito del módulo

Desarrollar aplicaciones de software, acorde con requerimientos del usuario y utilizando la metodología de programación orientada a objetos para agilizar y resolver problemas complejos relacionados con el procesamiento de información.

2.3 Mapa del Módulo

Nombre del Módulo	Unidad de Aprendizaje	Resultado de aprendizaje
<p>Programación orientada a objetos</p> <p>144 horas</p>	<p>1. Manejo de componentes del lenguaje orientados a objetos.</p> <p>60 horas</p>	<p>1.1 Identifica los componentes fundamentales de la programación orientada a objetos.</p> <p>16 horas</p>
		<p>1.2 Modela y codifica programas de cómputo haciendo uso del paradigma orientado a objetos, a través del uso de sentencias de control, objetos y clases.</p> <p>44 horas</p>
	<p>2. Desarrollo de aplicaciones orientadas a objetos.</p> <p>84 horas</p>	<p>2.1 Elabora aplicaciones mediante la interacción de los objetos y actores del sistema.</p> <p>60 horas</p>
		<p>2.2 Asegura la integridad de los datos implementando medidas de seguridad haciendo uso de patrones de diseño.</p> <p>24 horas</p>

2.4 Unidades de Aprendizaje

Unidad de aprendizaje:	1. Manejo de componentes del lenguaje orientados a objetos.	60 horas
Propósito de la unidad	Identificar los principales componentes de la programación orientada a objetos, con ellos modelará y codificará programas que le permitirán el desarrollo de aplicaciones o sistemas de información.	
Resultado de aprendizaje:	1.1 Identifica los componentes fundamentales de la programación orientada a objetos	16 horas

Actividades de evaluación	Evidencias a recopilar	Ponderación	Contenidos
<p>1.1.1 Determina la importancia del paradigma de la programación orientada a objetos a través del conocimiento de los principales componentes de la POO y la importancia de su modelado; elabora un portafolio de evidencias donde incluye:</p> <ul style="list-style-type: none"> • Documento de diseño de modelos bajo el enfoque de la programación orientada a objetos. • Tabla descriptiva de características de lenguajes de programación orientado a objeto. • Diagramas de modelos para el diseño orientado a objeto. 	<ul style="list-style-type: none"> • Portafolio de evidencias. 	25%	<p>A. Identificación de componentes de la programación orientada a objetos.</p> <ul style="list-style-type: none"> • Concepto de la programación orientada a objetos. • Objetos. <ul style="list-style-type: none"> – Definición. – Identificación de objetos. – Comunicación entre objetos. – Mensajes. – Paso de mensajes. – Estructura interna de un objeto. • Características de la programación orientada a objetos. <ul style="list-style-type: none"> – Abstracción. – Herencia. – Encapsulamiento. – Polimorfismo. • Clases. <ul style="list-style-type: none"> – Relación entre clases y objetos.

Actividades de evaluación	Evidencias a recopilar	Ponderación	Contenidos
			<p>B. Uso del lenguaje de modelado unificado para el diseño de programas orientados a objetos</p> <ul style="list-style-type: none"> • Objetos. • Clases. • Estados. • Secuencia. • Casos de uso. <p>C. Software para el desarrollo y modelado de aplicaciones orientada a objetos.</p> <ul style="list-style-type: none"> • Software de diagramación UML • Lenguajes de programación. <ul style="list-style-type: none"> – Plataforma. – Editor de texto empleado. – Compilador. – Enlazador. – Entornos de desarrollo integrados. • Sintaxis. • Herramientas de compilación.

Resultado de aprendizaje:	1.2 Modela y codifica programas de cómputo haciendo uso del paradigma orientado a objetos, a través del uso de sentencias de control, objetos y clases		44 horas
Actividades de evaluación	Evidencias a recopilar	Ponderación	Contenidos
<p>1.2.1 Desarrolla aplicaciones implementando los componentes de la programación orientada a objetos, donde aplica:</p> <ul style="list-style-type: none"> • Funciones de entrada y salida. • Sentencias de control. • Manejo de arreglos y estructuras. • Clases y objetos. 	<ul style="list-style-type: none"> • Documento con la implementación de la sintaxis de cada uno de los componentes de la programación orientada a objetos. • Problema solucionado con el programa propuesto. 	<p>25%</p>	<p>A. Manejo de funciones de entrada y salida.</p> <ul style="list-style-type: none"> • Datos numéricos. • Caracteres. • Formato. <p>B. Manejo de Sentencias de Control.</p> <ul style="list-style-type: none"> • Selección. <ul style="list-style-type: none"> – <i>If then else.</i> – <i>Switch.</i> • Repetición. <ul style="list-style-type: none"> – <i>Do while.</i> – <i>For.</i> – <i>While</i> <p>C. Programación con arreglos y estructuras.</p> <ul style="list-style-type: none"> • Arreglos unidimensionales. • Arreglos multidimensionales. • Manejo estructuras. <p>D. Programación de objetos y clases.</p> <ul style="list-style-type: none"> • Definición de una clase. <ul style="list-style-type: none"> – Atributos – Métodos • Control de acceso a los miembros de la clase. <ul style="list-style-type: none"> – Público – Privado – Protegido • Implementación de una clase

Actividades de evaluación	Evidencias a recopilar	Ponderación	Contenidos
			<ul style="list-style-type: none"> • Creación y eliminación de objetos <ul style="list-style-type: none"> – Constructor. – Destructor. – Asignación de objetos. – Referencias a objetos e instancias. • Funciones de objetos.
<p>Sesión para recapitulación y entrega de evidencias.</p>			

Unidad de aprendizaje:	2. Desarrollo de aplicaciones orientadas a objetos.	84 horas
Propósito de la unidad	Diseñar aplicaciones basadas en la metodología de la programación orientada a objetos a fin de obtener información procesada con el uso de objetos.	
Resultado de aprendizaje:	2.1 Elabora aplicaciones mediante la interacción de los objetos y actores del sistema.	60 horas

Actividades de evaluación	Evidencias a recopilar	Ponderación	Contenidos
<p>2.1.1 Desarrolla programas de cómputo utilizando las propiedades de herencia y polimorfismo.</p>	<ul style="list-style-type: none"> Documento que presente la codificación de cada una de las aplicaciones utilizando herencia y polimorfismo. 	<p>35 %</p>	<p>A. Identificación de las propiedades de la herencia.</p> <ul style="list-style-type: none"> Reutilización de código. Extensibilidad. Tipos de herencia <ul style="list-style-type: none"> Simple Múltiple Ambigüedad <p>B. Implementación de la herencia en objetos.</p> <ul style="list-style-type: none"> Simple Múltiple <p>C. Uso de sobrecarga de operadores</p> <ul style="list-style-type: none"> Operadores Unario Operadores Binarios <p>D. Implementación de polimorfismo</p> <ul style="list-style-type: none"> En funciones En funciones virtuales de objetos
<p>Sesión para recapitulación y entrega de evidencias.</p>			

Resultado de aprendizaje:	2.2 Asegura la integridad de los datos implementando medidas de seguridad haciendo uso de patrones de diseño.	24 horas	
Actividades de evaluación	Evidencias a recopilar	Ponderación	Contenidos
<p>2.2.1 Implementa medidas de seguridad en los programas diseñados haciendo uso de patrones de diseño.</p>	<ul style="list-style-type: none"> • Documento que presente la codificación de la aplicación desarrollada que asegure la integridad de los objetos. 	<p>15%</p>	<p>A. Identificación de patrones de diseño para la autenticación en aplicaciones.</p> <ul style="list-style-type: none"> • Fábricas abstractas. • Singleton o Singular. • Factory Method o Método de Fábrica. • Prototipo. • Implementación de los distintos patrones de diseño para la autenticación. <p>B. Aplicación de medidas para la operación continua del sistema.</p> <ul style="list-style-type: none"> • Cifrado de datos. • Inicialización de objetos. • Destrucción de objetos. • Manejo de excepciones.
<p>Sesión para recapitulación y entrega de evidencias.</p>			

2.5 Referencias

Básicas:

- Ceballos, F. (2015). *C/C++ - Curso de programación* (4ª ed.). RA-MA.
- Joyanes, L. (1996). *Programación Orientada a Objetos*. McGraw-Hill.
- Schmüller, J. (2001). *Aprendiendo UML en 24 horas*. Prentice Hall.
- López, J. y Gutiérrez, A. (2014). *Programación orientada a objetos con C++ y Java. Un intercambio interdisciplinario*. Patria.
- López, L. (2011). *Programación estructurada y orientada a objetos* (3ª ed.). Alfaomega.
- Blancarte, O. (2016). *Introducción a los patrones de diseño. Un enfoque práctico*. Oscar Blancarte Blog.
- Ceballos, F. (2015). *Java. Interfaces gráficas y aplicaciones para internet* (4ª ed.). RA-MA.
- García, A. (2017). *Java para niñ@s: Aprende a programar y crea tus propios proyectos con Java 8*. Independently published.

Complementarias:

- Kimmel, P. (2008). *Manual UML*. McGraw-Hill. Interamericana.
- Ceballos, F. (2007). *Programación Orientada a Objetos con C++* (4ª ed.). RA-MA.
- Marshal, D. (2008). *Programming Microsoft® Visual C#® 2008: The Language*. Microsoft.

Páginas Web:

- Ruiz, R. (s.f). *Fundamentos de Programación Orientada a Objetos*. Consultado el 03/06/2024. <https://goo.gl/81KtA5>
- Draw.IO. (s.f). *Draw.IO: Diagrama sin título*. Consultado el 03/06/2024. <https://www.draw.io/>
- ArgoUML. (s.f) *ArgoUML*. Consultado el 03/06/2024. <https://argouml-tigris-org.github.io/tigris/argouml/>
- Microsoft. (s.f.). *Introducción a lenguaje* Consultado el 03/06/2024. <https://learn.microsoft.com/es-es/training/modules/csharp-write-first/1-introduction?ns-enrollment-type=learningpath&ns-enrollment-id=learn.www.get-started-c-sharp-part-1>
- UML. (s.f.) *UML*. Consultado el 03/06/2024. <http://www.uml.org/>
- Microsoft. (s.f) *Documentación de Microsoft C++, C y Assembler*. Consultado el 03/06/2024. <https://learn.microsoft.com/es-mx/cpp/?view=msvc-170>