

1. Datos Generales de la asignatura

Nombre de la asignatura:	Tecnologías de la Información
Clave de la asignatura:	PSC-1037
SATCA¹:	2-2-4
Carrera:	Ingeniería en Pesquerías

2. Presentación

Caracterización de la asignatura
<p>La asignatura de Tecnología de la Información corresponde al área de Ingeniería en Pesquerías. Se imparte en el primer semestre, siendo esta una materia obligatoria cuya aportación al perfil del egresado permite la elaboración de programas en un lenguaje de alto nivel, para la solución de problemas de la Pesca y Acuicultura.</p> <p>Favorecer el desarrollo de una metodología sistemática de trabajo para el diseño y gestión de proyectos de investigación y desarrollo tecnológico que contribuyan en la innovación de sistemas de pesca y acuícolas.</p> <p>El objetivo de la materia es Conocer y comprender los fundamentos y estado actual de la informática, utilizando tanto el software de aplicación como el desarrollo de algoritmos computacionales codificados en un lenguaje de alto nivel, para la solución de problemas. El alumno pueda elaborar programas informáticos y aplicar el uso de paquetería de programas informáticos para apoyar a la solución de los problemas que se le presenten en el desarrollo de las pesquerías y la acuicultura.</p> <p>Que el alumno pueda elaborar e interpretar en forma tradicional a través de paquetes informáticos programas para apoyar los procesos de producción. Apoya a las materias de Ecología, Oceanografía, Recursos Pesqueros, Introducción a los sistemas de producción, Introducción a la tecnología pesquera, Electricidad y magnetismo Formulación y Evaluación de Proyectos, Procesos Administrativos, así como en la de Sistemas de Información Geográfica y Sensores Remotos, en las áreas de estudios previos, formulación y elaboración del estudio técnico y en la de fundamentos de procesos de producción pesqueros y acuícolas. Al finalizar la asignatura el educando será capaz de trabajar como asistente en cualquier despacho de ingeniería coadyuvando en la elaboración de programas computacionales que apoyen los proyectos productivos y de investigación en las áreas de pesquerías y acuicultura.</p>
Intención didáctica
<p>Utilizar equipo de cómputo mediante procedimientos y algoritmos que permitan diseñar y elaborar programas que apoyen la elaboración de proyectos de producción, y de investigación científica que se apliquen en la evaluación del impacto que tienen las pesquerías y la acuicultura en el medio ambiente, que facilite el manejo de los procesos de</p>

¹ Sistema de Asignación y Transferencia de Créditos Académicos

la producción, que ayude a agilizar el cálculo de costos de cualquier proyectos productivo que se opere en las pesquerías.

Se organiza el temario, en seis unidades, agrupando los contenidos en la primera unidad se hace una introducción a la informática donde se su importancia lo que es el hardware, software, sistemas operativos, lenguajes de programación, redes Internet y más detalles.

En las unidades dos y tres se aborda las aplicaciones de los sistemas operativos, programas y el ambiente de desarrollo.

Las unidades cuatro, cinco y seis trabaja con la programación tanto en sus funciones y procedimientos y estructuras de control así como en arreglos y archivos que resuelvan problemas aplicados a la actividad pesquera y acuícola, que se presenten en el entorno regional.

La idea es abordar reiteradamente los conceptos fundamentales hasta conseguir su comprensión. Se propone reforzar con ejercicios prácticos aplicando paquetería informática en la elaboración de programas durante el desarrollo de todo el curso, se requiere contar con el apoyo del centro de cómputo y una computadora por alumno con Internet para la realización de las búsquedas informáticas de los diferentes conceptos y normas requeridos para el desarrollo del curso

Se sugiere una actividad integradora, en todas las unidades, que permita aplicar los conceptos informáticos estudiados. Esto permite dar un cierre a la materia mostrándola como útil por sí misma en el desempeño profesional, independientemente de la utilidad que representa en el tratamiento de temas en materias posteriores.

El enfoque sugerido para la materia requiere que las actividades prácticas promuevan el desarrollo de habilidades para la experimentación, tales como: identificación, manejo y control de variables y datos relevantes; planteamiento de hipótesis; trabajo en equipo; asimismo, propicien procesos intelectuales como inducción-deducción y análisis-síntesis con la intención de generar una actividad intelectual compleja; por esta razón varias de las actividades prácticas se han descrito como actividades previas al tratamiento teórico de los temas, de manera que no sean una mera corroboración de lo visto previamente en clase, sino una oportunidad para conceptualizar a partir de lo observado. En las actividades prácticas sugeridas, es conveniente que el profesor busque sólo guiar a sus alumnos para que ellos hagan la elección de las variables a controlar y registrar. Para que aprendan a planificar, que no planifique el profesor todo por ellos, sino involucrarlos en el proceso de planeación.

En las actividades de aprendizaje sugeridas, generalmente se propone la formalización de los conceptos a partir de experiencias concretas; se busca que el alumno tenga el primer contacto con el concepto en forma concreta y sea a través de la observación, la reflexión y la discusión que se dé la formalización; la resolución de problemas se hará después de este proceso. Esta resolución de problemas no se especifica en la descripción de actividades, por ser más familiar en el desarrollo de cualquier curso. Pero se sugiere que el alumno utilice los programas computacionales desde el principio y ejercite en la elaboración de programas aplicados a las pesquerías y la acuicultura.

En el transcurso de las actividades programadas es muy importante que el estudiante aprenda a valorar las actividades que lleva a cabo y entienda que está construyendo su

hacer futuro y en consecuencia actúe de una manera profesional; de igual manera, aprecie la importancia del conocimiento y los hábitos de trabajo; desarrolle la precisión y la curiosidad, la puntualidad, el entusiasmo y el interés, la tenacidad, la flexibilidad y la autonomía.

3. Participantes en el diseño y seguimiento curricular del programa

Lugar y fecha de elaboración o revisión	Participantes	Evento
Instituto Tecnológico de Mazatlán del 23 al 27 de noviembre de 2009.	Representantes de los Institutos Tecnológicos de: Boca del Río, Guaymas, Mazatlán y Salina Cruz.	Reunión Nacional de Diseño e Innovación Curricular para el Desarrollo y Formación de Competencias Profesionales de las Carreras de Ingeniería en Acuicultura, Ingeniería en Pesquerías e Ingeniería Naval.
Instituto Tecnológico de Boca del Río del 26 al 30 de abril de 2010.	Representantes de los Institutos Tecnológicos de: Boca del Río, Guaymas y Mazatlán.	Reunión Nacional de Consolidación de los Programas en Competencias Profesionales de Ingeniería en Acuicultura, Ingeniería en Pesquerías e Ingeniería Naval.
Tecnológico Nacional de México, del 26 al 30 de agosto de 2013.	Representantes de los Institutos Tecnológicos de: Boca del Río, Guaymas, Lerma y Mazatlán.	Reunión Nacional de Seguimiento Curricular de las carreras de Ingeniería en Nanotecnología, Ingeniería Petrolera, Ingeniería en Acuicultura, Ingeniería en Pesquerías, Ingeniería Naval y Gastronomía del SNIT.

4. Competencia(s) a desarrollar

Competencia(s) específica(s) de la asignatura
Conoce, comprende y aplica los fundamentos y estado actual de la informática, para la solución de problemas y cálculos propios de la ingeniería en Pesquerías, utilizando tanto el software de aplicación como el desarrollo de algoritmos computacionales codificados en un lenguaje de alto nivel,

5. Competencias previas

Manejar y aplicar el ambiente Windows y Office.

Analizar y sintetizar información para comprender los principios y fundamentos de la programación.

Organizar y planificar el plan de trabajo para la elaboración de los Algoritmos.

Redactar informes técnicos en su propia lengua de manera oral y escrita.

Emplear los comandos de los programas compiladores de una segunda lengua,

6. Temario

Unidad	Temas	Subtemas
1	Introducción a la Informática.	<p>1.1 Importancia y perspectivas de la computación.</p> <p>1.2 Impacto de las computadoras en la sociedad.</p> <p>1.3 Hardware.</p> <p>1.3.1 CPU.</p> <p>1.3.2 Periféricos.</p> <p>1.4 Software.</p> <p>1.4.1 Sistemas Operativos.</p> <p>1.4.2 Lenguajes de Programación: Compiladores.</p> <p>1.4.2.1 Alto nivel: Visual Basic, FORTRAN 90, Pascal, entre otros.</p> <p>1.4.2.2 Bajo nivel: Ensamblador. Hexadecimal, binario.</p> <p>1.4.3 Programas de aplicación: Principales aplicaciones: procesadores de texto, hojas electrónicas, manejadores de bases de datos, diseñadores de presentaciones, graficadores, software científico y de ingeniería entre otros.</p> <p>1.5 Maneras de operación de la computadora: Interrelación usuario-computadora.</p> <p>1.6 Redes.</p> <p>1.7 Internet.</p> <p>1.7.1 Características e importancia.</p> <p>1.7.2 Aplicaciones: ftp, telnet, www, entre otras.</p> <p>1.7.3 Búsqueda de información. Robots buscadores.</p> <p>1.7.4 Seguridad de la información.</p> <p>1.7.4.1 Antivirus, Firewall.</p> <p>1.7.4.2 Spywares.</p>

2	Sistemas operativos y programas de Aplicación	2.1 Entorno Windows: 2.1.1 Archivos y carpetas. 2.1.2 Manejo de información. 2.2 Entorno UNIX (Linux). 2.3 Procesadores de textos: Características y aplicaciones. 2.4 Diseñadores de presentaciones: Características y aplicaciones. Estrategias de desarrollo de presentaciones. 2.5 Hojas electrónicas. 2.5.1 Características y aplicaciones 2.5.2 Funciones de librería 2.5.3 Graficación de datos
3	El ambiente de desarrollo de aplicaciones.	3.1 Desarrollo histórico de un lenguaje estructurado. 3.2 Desarrollo y características de los lenguajes visuales. 3.3 El ambiente de desarrollo de la aplicación. 3.3.1 Descripción de los menús y las opciones de ambiente de desarrollo de aplicaciones. 3.4 Estructura general de una aplicación. 3.4.1 Interfase y código. 3.4.1.1 Proyecto. 3.4.1.2 Formularios. 3.4.1.3 Módulos de código. 3.5 Eventos. 3.6 Objetos y propiedades. 3.7 Herramientas para crear la interfase de la aplicación. 3.7.1 Formularios y módulos de código. 3.7.2 Cajas de texto. 3.7.3 Etiquetas. 3.7.4 Botones de comando. 3.7.5 Botones de opción. 3.7.6 Casillas de verificación. 3.7.7 Cuadros de imagen. 3.7.8 Otras herramientas.
4	Introducción a la programación.	4.1 Metodología de la solución de problemas con la computadora. 4.1.1 Programación no estructurada 4.1.2 Programación estructurada 4.1.3 Programación modular 4.1.4 Programación orientada a eventos 4.1.5 Programación orientada a objetos. 4.1.6 Programación por escenarios

		<p>4.2 Algoritmos</p> <p>4.2.1 Diagramas de bloque, flujo, simbología</p> <p>4.2.2 Pseudocódigo y codificación</p> <p>4.3 Elementos del lenguaje de programación</p> <p>4.3.1 Palabras reservadas</p> <p>4.3.2 Tipos de datos básicos</p> <p>4.3.3 Declaración de variables, constantes, procedimientos y funciones (subrutinas)</p> <p>4.3.4 Operaciones aritméticas, lógicas y relacionales.</p> <p>4.4 Uso del depurador (debugger) para depurar y dar seguimiento a los programas.</p> <p>4.5 Diseño, edición, compilación y ejecución de aplicaciones.</p>
5	Funciones, procedimientos y estructuras de control.	<p>5.1 Funciones y procedimientos.</p> <p>5.1.1 Funciones estándar</p> <p>5.2 Funciones definidas por el usuario.</p> <p>5.2.1 Paso de parámetros por valor.</p> <p>5.2.2 Paso de parámetros por referencia.</p> <p>5.3 Estructuras de selección.</p> <p>5.3.1 Selectiva simple.</p> <p>5.3.2 Selectiva doble.</p> <p>5.3.3 Selectiva anidada.</p> <p>5.3.4 Selectiva múltiple.</p> <p>5.4 Estructuras Repetitivas (Bucles).</p> <p>5.4.1 Repetir mientras.</p> <p>5.4.2 Repetir hasta.</p> <p>5.4.3 Repetir desde.</p> <p>5.5 Ejercicios de aplicación.</p>
6	Arreglos, Archivos	<p>6.2 Archivos.</p> <p>6.2.1 Archivos secuenciales.</p> <p>6.2.2 Archivos de acceso directo.</p> <p>6.2.3 Abrir, cerrar, acceder y guardar información en archivos.</p> <p>6.3 Problemas de aplicación.</p>

7. Actividades de aprendizaje de los temas

Introducción a la Informática	
Competencias	Actividades de aprendizaje
<p>Competencia específica: Conoce el ambiente de desarrollo de la informática para la creación de aplicaciones y desarrollo de la capacidad de crear interfaces considerando aplicaciones de variada complejidad.</p> <p>Competencias genéricas: Capacidad de análisis y síntesis. Solución de Problemas. Habilidad para búsqueda de información. Capacidad para trabajar en equipo. Habilidad en el uso de tecnologías de información y comunicación. Capacidad de aplicar los conocimientos en la práctica. Comunicación oral y escrita.</p>	<ul style="list-style-type: none"> • Investigar en diversas fuentes sobre las tecnologías de la información. • Valorar los elementos que conforma un equipo computacional en software, hardware y Firmware. • Realizar una clasificación de los sistemas operativos, valorando sus actuales diferencias • Acceder al sistema ambiente para analizar y comprender los conceptos básicos de la paquetería de software • Elaborar un resumen y mapa conceptual • Exposición en sesión grupal
Sistemas operativos y programas de aplicación	
Competencias	Actividades de aprendizaje
<p>Competencia específica: Reconocer y aplicar las funciones básicas del sistema operativo y los programas de aplicación de la ofimática para la elaboración de reportes en un contexto académico científico</p> <p>Competencias genéricas: Capacidad de análisis y síntesis. Solución de Problemas. Habilidad para búsqueda de información. Capacidad para trabajar en equipo. Habilidad en el uso de tecnologías de información y comunicación. Capacidad de aplicar los conocimientos en la práctica. Comunicación oral y escrita.</p>	<ul style="list-style-type: none"> • Interactuar con equipo de computo • Demostrar la operatividad y manejo de equipo • Desarrollar actividades empleado la ofimática • Aplicación en reportes y materiales de exposición • Generar círculos de creatividad empleando la ofimática • Generar dinámicas de aplicación de la ofimática • Retroalimentar con las experiencias

El ambiente de desarrollo de aplicaciones.	
Competencias	Actividades de aprendizaje
<p>Competencia específica: Analiza, diferencia e interpreta el ambiente de desarrollo para la creación de aplicaciones e interfaces en un nivel de variada complejidad.</p> <p>Competencias genéricas: Capacidad de análisis y síntesis. Solución de Problemas. Habilidad para búsqueda de información. Capacidad para trabajar en equipo. Habilidad en el uso de tecnologías de información y comunicación. Capacidad de aplicar los conocimientos en la práctica. Comunicación oral y escrita.</p>	<ul style="list-style-type: none"> • Analizar estudios de caso de ambientes de aplicación y creación de interfaces • Identificar, definir y relacionar las variables de los ambientes de aplicación e interfaces • Gestionar información para instalar y configurar el lenguaje de programación a utilizar. • Estructurar las aplicaciones e interfaces • Elaborar un resumen con las opciones de cada menú del ambiente de desarrollo y su función. • Elaborar una tabla con los accesos rápidos a las funciones más importantes en el ambiente de desarrollo. • Acceder al sistema de ayuda del ambiente de desarrollo para resolver cualquier duda que se tenga sobre las herramientas o los comandos del lenguaje. • Crear las interfaces para aplicaciones que indique el profesor.
Introducción a la programación	
Competencias	Actividades de aprendizaje
<p>Competencia específica: Analiza y comprende el concepto de algoritmo y la terminología relacionada para identificar y aplicar las características de las técnicas de diseño y lenguaje algorítmico gráfico o manuscrito considerando la utilización de medios y códigos en un lenguaje de programación.</p> <p>Competencias genéricas: Capacidad de análisis y síntesis. Solución de Problemas. Habilidad para búsqueda de información.</p>	<ul style="list-style-type: none"> • Analizar estudios de caso de programación • Reconocer y comprender los conceptos fundamentales de la programación • Generar una estructura básica • Elaborar algoritmos de actividades cotidianas. • Organizar intercambios grupales acerca de conceptos de: identificador, constantes, variables y la proposición de asignación, entre otros.

Capacidad para trabajar en equipo. Habilidad en el uso de tecnologías de información y comunicación. Capacidad de aplicar los conocimientos en la práctica. Comunicación oral y escrita.	<ul style="list-style-type: none"> • Construir y evaluar expresiones matemáticas. • Compilar y ejecutar un programa modelo.
Funciones, procedimientos y estructuras de control	
Competencias	Actividades de aprendizaje
<p>Competencia específica: Analiza, comprende y aplica el uso de funciones y procedimientos en la programación modular para resolver problemas considerando las funciones, procedimientos y estructuras de control</p> <p>Competencias genéricas: Capacidad de análisis y síntesis. Solución de Problemas. Habilidad para búsqueda de información. Capacidad para trabajar en equipo. Habilidad en el uso de tecnologías de información y comunicación. Capacidad de aplicar los conocimientos en la práctica. Comunicación oral y escrita.</p>	<ul style="list-style-type: none"> • Examinar estudios de casos e identificar los procesos de resolución de problemas • Realizar ejemplos que requieran funciones estándar. • Realizar ejemplos utilizando funciones definidas por el usuario • Realizar ejemplos utilizando paso de parámetros por valor y por referencia. • Usar el depurador del ambiente de desarrollo para seguir el paso de los parámetros en las funciones y los procedimientos. • Realizar una síntesis sobre el funcionamiento y aplicación de las estructuras selectivas. • Implementar el diseño de programas que requieran estructuras selectivas para probarlas en una aplicación.
Arreglos, Archivos	
Competencias	Actividades de aprendizaje
<p>Competencia específica: Comprende y aplica el uso y funcionamiento de las estructuras de selección y de repetición para implementarlas en el diseño de aplicaciones considerando diversos arreglos y archivos</p> <p>Competencias genéricas: Capacidad de análisis y síntesis. Solución de Problemas. Habilidad para búsqueda de información. Capacidad para trabajar en equipo.</p>	<ul style="list-style-type: none"> • Generar ejercicios de aplicación que requieran: <ul style="list-style-type: none"> ○ funciones estándar. ○ funciones definidas por el usuario ○ utilizar paso de parámetros por valor y por referencia. • Usar el depurador del ambiente de desarrollo para seguir el paso de los

<p>Habilidad en el uso de tecnologías de información y comunicación. Capacidad de aplicar los conocimientos en la práctica. Comunicación oral y escrita.</p>	<p>parámetros en las funciones y los procedimientos.</p> <ul style="list-style-type: none"> • Realizar una síntesis sobre el funcionamiento y aplicación de las estructuras selectivas. • Implementar el diseño de programas que requieran estructuras selectivas para probarlas en una aplicación.
--	---

8. Práctica(s)

<ul style="list-style-type: none"> • El lenguaje de programación deberá definirse por el Instituto Tecnológico, aunque se sugiere emplear Visual Basic (en su forma completa o como <i>addendum</i> de Excel), Pascal o FORTRAN 90. • Elaboración de un programa para resolver un sistema de ecuaciones lineales por el método de Gauss.
--

9. Proyecto de asignatura

<p>El objetivo del proyecto que planteé el docente que imparta esta asignatura, es demostrar el desarrollo y alcance de la(s) competencia(s) de la asignatura, considerando las siguientes fases:</p> <ul style="list-style-type: none"> • Fundamentación: marco referencial (teórico, conceptual, contextual, legal) en el cual se fundamenta el proyecto de acuerdo con un diagnóstico realizado, mismo que permite a los estudiantes lograr la comprensión de la realidad o situación objeto de estudio para definir un proceso de intervención o hacer el diseño de un modelo. • Planeación: con base en el diagnóstico en esta fase se realiza el diseño del proyecto por parte de los estudiantes con asesoría del docente; implica planificar un proceso: de intervención empresarial, social o comunitario, el diseño de un modelo, entre otros, según el tipo de proyecto, las actividades a realizar los recursos requeridos y el cronograma de trabajo. • Ejecución: consiste en el desarrollo de la planeación del proyecto realizada por parte de los estudiantes con asesoría del docente, es decir en la intervención (social, empresarial), o construcción del modelo propuesto según el tipo de proyecto, es la fase de mayor duración que implica el desempeño de las competencias genéricas y específicas a desarrollar. • Evaluación: es la fase final que aplica un juicio de valor en el contexto laboral-profesión, social e investigativo, ésta se debe realizar a través del reconocimiento de logros y aspectos a mejorar se estará promoviendo el concepto de “evaluación para la mejora continua”, la metacognición, el desarrollo del pensamiento crítico y reflexivo en los estudiantes.
--

10. Evaluación por competencias

Para evaluar las actividades de aprendizaje se recomienda solicitar: mapas conceptuales, reportes de prácticas, estudios de casos, exposiciones en clase, ensayos, problemarios, reportes de visitas, portafolio de evidencias y cuestionarios.

Para verificar el nivel del logro de las competencias del estudiante se recomienda utilizar: listas de cotejo, listas de verificación, matrices de valoración, guías de observación, coevaluación y autoevaluación.

11. Fuentes de información

1. Jean Paul Tremblay & Richard B. Bunt. (1988). *Introducción a la ciencia de las computadoras (enfoque algorítmico)*. McGraw Hill.
2. Luis Joyanes Aguilar, Peña Tresancos Jaime, Vidal Fernández, María Carmen. (2004). *Introducción a la informática*. España: Mc. Graw Hill.
3. Levine, Guillermo. (1989). *Introducción a la computación y a la programación estructurada*. Ed. Mc. Graw Hill.
4. Seymour E. Goodman, S. T. Hedetniemi. (1997). *Introduction to the design and analysis of algorithms*. Mc. Graw Hill.
5. Joyanes Aguilar, Luis. (1996). *Fundamentos de programación, algoritmos y estructura de datos*. Mc. Graw Hill.
6. Forsythe, Queman. Organik, Stenberg.(1998). *Lenguajes de diagramas de flujo*. Limusa.
7. O. J. Dahl, E. W. Dijatra, C.A.R. Hoare. (1996). *Programación estructurada*. Tiempo contemporáneo
8. Simon Mochon Cohen. (2000). *Actividades para el Autoaprendizaje Dirigido de: Word, Excel y Powerpoint*. McGraw-Hill
9. Baase, Sara . (2002). *Algoritmos Computacionales*. Prentice Hall
10. Armando E. De Guisti. (2001). *Algoritmos, Datos y Programas*. Universidad Nacional de la Plata Prentice Hall.
11. Ruth Maran, Paul Whitehead. (2002) *Aprenda Internet y la WWW Visualmente*. Trejos Hermanos Sucesores.
12. Ramírez R, José Felipe. (2001). *Aprenda Visual Basic Practicando*. Pearson

Education.

13. Ceballos, Fco. Javier. (2000) *Curso de programación de Visual Basic 6*. México :Alfaomega,

14. Ceballos, F.J. (1996). *De QBasic a Visual Basic. Curso de Programación*. RA-MA