

Planificación anual por trimestre Informática Personal y Profesional

ESPACIO CURRICULAR:	Programación II
CURSO:	5 "E" y "F"
DOCENTE:	Araujo Oscar, Kolb, Mariela Elizabeth

FUNDAMENTACIÓN

En este espacio curricular, los estudiantes podrán escribir algoritmos utilizando estructuras más complejas que permitirán mayor eficiencia en la resolución de sus problemas. Verificar y depurar el producto desarrollado para asegurarse que cumple con las especificaciones recibidas.

Se continúa con lo desarrollado en el espacio curricular del mismo nombre durante el año anterior, Programación I. Se busca la adquisición de más experiencia, en cuanto a técnicas, del lenguaje utilizado en esa oportunidad (PHP).

PROPÓSITOS

Tomar el hábito de la diagramación, pruebas de escritorio y otras técnicas, para luego codificar con seguridad, como así también, optimizar código mediante "buenas prácticas". Solucionar problemas complejos con la ayuda de grupos y docentes.

OBJETIVOS

- Que los alumnos sean capaces de resolver de manera eficaz distintos problemas de carácter general, con independencia del lenguaje de programación utilizado. Este objetivo requiere que los alumnos adquieran el conocimiento teórico y práctico sobre las técnicas básicas de programación estructurada y orientada a objetos logrando la *obtención de ideas intuitivas y claras de los conceptos y técnicas estudiados* facilitando la aplicación práctica de los algoritmos.

COMPETENCIAS**A) BÁSICAS**

- RESUELVE problemas complejos mediante algoritmos basados en el paradigma estructurado y orientado a objetos.
- OPTIMIZA el código de programación aplicando las buenas prácticas.

B) ESPECÍFICAS

- TOMA el hábito de las buenas prácticas para codificar.
- VALORA la importancia de la documentación de su código.
- REALIZA las pruebas necesarias para evitar errores en tiempo de ejecución.
- INTEGRA eficientemente bases de datos en sus soluciones.
- COMPRENDE los conceptos básicos de la programación orientada a objetos (POO).
- APLICA la POO para resolver problemas prácticos.

CONTENIDOS

PRIMER TRIMESTRE	CAPACIDADES	ACTIVIDADES	INDICADORES/ EVIDENCIAS DE DESEMPEÑO
<p>Revisión y repaso de temas de programación estructurada, mediante prácticas usando lenguaje PHP (complementadas con HTML-CSS-JS):</p> <ul style="list-style-type: none"> • Arreglos unidimensionales (vectores) y bidimensionales (matrices). • Declaración. • Métodos de búsqueda y ordenamiento. • Arreglos de caracteres. 	<ul style="list-style-type: none"> ▪ Domina el uso de vectores y matrices (PHP). ▪ Comprende cómo implementar funciones para simplificar problemas complejos. ▪ Comprende cómo acceder archivos (PHP). ▪ Comprende cómo conectar una base de datos según el tipo de conexión. 	<p>Mediante caso práctico:</p> <ul style="list-style-type: none"> ▪ Crear un algoritmo que utilice vectores. Codificarlo en PHP. Hacer pruebas para depurar posibles errores. ▪ Crear un algoritmo que utilice matrices. Codificarlo en PHP. Hacer pruebas para depurar posibles errores. <p>Mediante caso práctico:</p> <ul style="list-style-type: none"> ▪ Crear algoritmo que implemente funciones. Codificarlo en PHP. Hacer 	<ul style="list-style-type: none"> ▪ Resuelve satisfactoriamente ejercicio práctico implementando vectores. ▪ Resuelve satisfactoriamente ejercicio práctico implementando matrices. ▪ Resuelve satisfactoriamente ejercicio práctico implementando funciones. ▪ Resuelve satisfactoriamente ejercicio práctico implementando archivos.

<ul style="list-style-type: none"> • Funciones: Ventajas de dividir la problemática en problemas menores. • Pasaje de parámetros por valor y por referencia. • Procesamiento de archivos. • Acceso a bases de datos: conexión. Tipos de conexiones. 		<p>pruebas para depurar posibles errores.</p> <p>Mediante caso práctico:</p> <ul style="list-style-type: none"> ▪ Crear algoritmo que implemente el uso de archivos. Codificarlo en PHP. Hacer pruebas para depurar posibles errores. <p>Mediante caso práctico:</p> <ul style="list-style-type: none"> ▪ Codificar el acceso a una base de datos en PHP. Hacer pruebas para depurar posibles errores. 	<ul style="list-style-type: none"> ▪ Resuelve satisfactoriamente ejercicio práctico para conectar bases de datos.
SEGUNDO TRIMESTRE	CAPACIDADES	ACTIVIDADES	INDICADORES/ EVIDENCIAS DE DESEMPEÑO
<ul style="list-style-type: none"> • Operaciones de manipulación de datos en una base de datos: insertar, modificar, borrar y/o consultar registros. • Acceso a datos de una base de datos mediante permisos. Registrar un nuevo usuario y/o acceder a una cuenta de usuario registrado. • Uso de Sesiones en PHP. • Paradigma Orientado a Objetos: introducción. • Clases vs. Objetos. Ciclo de vida de los objetos. 	<ul style="list-style-type: none"> ▪ Comprende cómo insertar un nuevo registro, modificar y eliminar un registro existente en una base de datos. ▪ Comprende cómo realizar autenticación de usuarios y permisos, usando base de datos. ▪ Comprende el uso de sesiones en aplicaciones que requieren autenticación de acceso de usuarios. ▪ Comprende los conceptos de la programación orientada a objetos. 	<p>Mediante caso práctico:</p> <ul style="list-style-type: none"> ▪ Mediante el paradigma estructurado, realizar el código para el registro de nuevo usuario y posterior autenticación de acceso a panel de acciones CRUD (crear, leer, actualizar, borrar) sobre tablas. ▪ Convertir el caso práctico de Registro- Autenticación+CRUD, utilizando el paradigma orientado a objetos. 	<ul style="list-style-type: none"> ▪ Resuelve satisfactoriamente ejercicio práctico implementando sesiones. ▪ Resuelve satisfactoriamente ejercicio práctico implementando lo abordado de manera integral: bases de datos, archivos, sesiones. ▪ Resuelve satisfactoriamente ejercicio práctico implementando el enfoque orientado a objetos: uso de Clases y Objetos, propiedades y métodos.

<ul style="list-style-type: none"> • Propiedades de un objeto. 	<ul style="list-style-type: none"> ▪ Comprende cómo y cuando crear/definir clases y objetos. ▪ Comprende cómo implementar propiedades de un objeto. 		
TERCER TRIMESTRE	CAPACIDADES	ACTIVIDADES	INDICADORES/ EVIDENCIAS DE DESEMPEÑO
<ul style="list-style-type: none"> • Métodos de acceso. Calificadores de los métodos. Nivel de acceso de métodos y propiedades. • Conceptos de herencia. Interfaces. Clases abstractas. Polimorfismo. • Aplicación de paradigma orientado a objetos en Proyecto Integrador. 	<ul style="list-style-type: none"> ▪ Comprende cómo implementar métodos en un objeto. ▪ Comprende cómo implementar herencia y polimorfismo. 	<p>Mediante trabajo práctico integrador:</p> <ul style="list-style-type: none"> • Implementar lo abordado sobre el paradigma orientado a objetos. 	<ul style="list-style-type: none"> ▪ Desarrolla satisfactoriamente una aplicación para la gestión contable que responda los criterios solicitados por los docentes de las asignaturas implicadas en el trabajo.

PROPUESTA METODOLÓGICA PARA LA ENSEÑANZA

Inicialmente, se hace una revisión y repaso de los temas vistos en Programación I para reforzarlos ya que se articulan con temas a desarrollarse en Programación II. Luego, cada unidad temática se aborda presentado al alumno una perspectiva teórica de los conceptos básicos involucrados para luego exponer ejemplos prácticos que permiten desarrollar los aspectos avanzados. También, se realiza la aplicación de los conceptos adquiridos en ejercicios prácticos que los alumnos desarrollan con la asistencia de los docentes. Finalmente, se plantean problemas, de creciente nivel de complejidad y carácter integrador, cuya resolución involucra la aplicación del total de temas vistos en la materia. En todo momento, se busca instaurar un ámbito crítico y reflexivo tanto de los conocimientos previamente adquiridos como también de los presentados en ese momento.

INSTRUMENTOS DE EVALUACIÓN

La evaluación del alumno es permanente.

Se utilizarán como instrumentos de evaluación:

- T.P.O.s: Realización de Ejercicios Prácticos obligatorios dónde la cantidad de líneas de código para resolver un problema es la unidad de calificación (50 líneas = 😊, >50 líneas = 😐, >100 líneas = 😞),
- Exposición Oral,
- Evaluación Escrita,
- Realización de Trabajo Integrador (interdisciplinar) a elección de los alumnos (en caso de grupos, hasta 5 integrantes) utilizando el lenguaje PHP, donde intervenga como funcionalidad:
 - la gestión contable,
 - acciones CRUD,
 - registro y autenticación de usuarios (manejo de sesiones).

FIRMA DE LOS DOCENTES

PROGRAMA

UNIDAD N°1: Revisión y repaso de temas de programación estructurada, mediante prácticas usando lenguaje PHP (complementadas con HTML-CSS-JS):

- Arreglos unidimensionales (vectores) y bidimensionales (matrices).
- Declaración.
- Métodos de búsqueda y ordenamiento.
- Arreglos de caracteres.
- Funciones: Ventajas de dividir la problemática en problemas menores.
- Pasaje de parámetros por valor y por referencia.
- Procesamiento de archivos.

UNIDAD N°2: Incorporación de Bases de Datos y uso de SQL.

- Acceso a bases de datos: conexión. Tipos de conexiones.
- Operaciones de manipulación de datos en una base de datos (CRUD): insertar, modificar, borrar y/o consultar registros.
- Acceso a datos de una base de datos mediante permisos. Registrar un nuevo usuario y/o acceder a una cuenta de usuario registrado.
- Uso de Sesiones en PHP.

UNIDAD N°3: Paradigma Orientado a Objetos

- Introducción a la POO
- Clases vs. Objetos. Ciclo de vida de los objetos.
- Propiedades de un objeto.
- Métodos de acceso. Calificadores de los métodos. Nivel de acceso de métodos y propiedades.
- Conceptos de herencia. Interfaces. Clases abstractas. Polimorfismo.
- Aplicación de paradigma orientado a objetos en Proyecto Integrador.

BIBLIOGRAFÍA

- Programación orientada a objetos en PHP, AVA - Formación en Ambientes Virtuales de Aprendizaje - SENA - Servicio Nacional de Aprendizaje.
- Nixon, R. (2015). Learning PHP, MySQL & Javascript, Fourth Edition. Cambridge: O'reilly
- <https://www.php.net/manual/es/index.php>.