



**UNIVERSIDAD TECNOLÓGICA  
FACULTAD DE INGENIERÍA EN SISTEMAS COMPUTACIONALES  
CAMPUS VICTOR LEVIS SASSO**

**INGENIERÍA WEB**

 **Guía de Laboratorio – Mini Proyecto #2  
de Sentencias de Control y Clases**

**INSTRUCTOR:** Ing. Irina Fong

Grupo:

Fecha de Entrega:

**Fecha Límite de entrega: [SOLO EN ESTE PERÍODO]**

**Del 22 al 03 de septiembre de 2025.**

Nota de Charla	
Estudiante #1	Estudiante #2

Nota de Investigación y Código Fuente	
Estudiante #1	Estudiante #2

**A.  Título de la experiencia**

**“Resolviendo problemas con estructuras de decisión y repetición en PHP”**

**B.  Tema**

Sentencias de control (if, case), operadores ternarios y estructuras repetitivas (while, for, foreach), arreglos, funciones y clases.

**C.  Objetivo**

**Objetivo General**

Construir aplicaciones web aplicando técnicas, metodologías y herramientas de diseño y desarrollo que permita la optimización, facilidad de mantenimiento, cumpliendo los criterios de usabilidad y accesibilidad.

Elaborado por: Ing. Irina Fong

13

Página 1 de

### **Objetivos Específicos:**

Aplicar estructuras de control condicional y repetitiva, funciones matemáticas, funciones de validación, clases con métodos estáticos, para resolver problemas algorítmicos, utilizando buenas prácticas de programación como las recomendaciones de PSR-1.

## D. Metodología

En Grupo de dos estudiantes resolverán una serie de problemas que requieren el uso de estructuras como if, case, while, for, foreach, switch, arreglos, funciones y clases. Cada problema deberá resolverse mediante el lenguaje de programación PHP. Se evaluará no solo la lógica del algoritmo, sino también su claridad, estilo y nomenclatura utilizada (**camelCase** para nombres de métodos/procedimientos, camelCase para variables).

### (PSR-1: Basic Coding Standard (Status: Accepted)

Este estándar comprende como debería ser la codificación de los elementos necesarios para asegurar la interoperabilidad de código compartido. Es decir, indican normas de codificación básica a muy bajo nivel.

1. **clases (StudyCaps)** o métodos (**camelCase**).  
**Ejemplo:**  

```
class UsuarioInvalidoException {}  
class GestorDeSesiones {}
```
2. **Métodos** [Deben declararse en camelCase]  

```
public function imprimirError() {}  
public function obtenerDatosUsuario() {}
```
3. **Constantes**  

```
const VERSION_APP = '1.0.0';  
const MAX_INTENTOS = 5
```
4. **Funciones y variables** (Aunque PSR-1 no obliga, por convención se recomienda )  

```
$notaFinal = 85;  
$listaEstudiantes = [];
```



## ✿ Entregables (Comprimido ZIP,RAR en Moodle)

Informe o Memoria  
del Trabajo

Enlace del repositorio.  
Ubicar Taller\_  
Siglasde los Integrantes.

Power Point u otro  
formato de  
presentación.

### 1. Informe o memoria del trabajo: PDF

El informe escrito debe detallar lo siguiente:

- Portada
- Índice
- Introducción
- Metodología
- Definición de Funciones Matemáticas, Cadenas y de Validación de Datos.
- Lecciones aprendidas o dificultades enfrentadas.
- Captura de pantalla de la ejecución de cada programa.
- Debe incluir números de páginas.

### 2. Código Fuente en PHP.

Facilitar el enlace del repositorio o subir la carpeta en Moodle comprimida. Documentar el repositorio. Ubicar en el Repositorio tecnologías utilizadas. Fecha de Realización, Estudiantes que pertenecen al grupo, correo, curso, y que digan que pertenecen a la Universidad Tecnológica de Panamá. Ubicar una Introducción, además si utilizaron POO, algún método estático, documentar algunas funciones matemáticas sqr, ect. Además funciones para validación y sanitización.

### 3. Presentación PowerPoint u otro formato de presentación.

- Una presentación con un mínimo de:

- Portada con datos de ambos estudiantes.
- Descripción general del taller.
- Descripción del Problema (con capturas del código).
- Capturas de Pantallas de los Problemas con sus soluciones.
- Diseño visual limpio y coherente.
- Imágenes claras y visibles.
- Capturas de códigos claros y visibles.
- Conclusiones o reflexiones del equipo.

3. Demostración en vivo (Explicar su código fuente con claridad).

Ejecución al azar de 4 problemas.

4. Utilización oportuna de las estructuras de control repetitivas. Utilice switch.

<https://codersfree.com/posts/operadores-ternarios-en-php>

6. Se tomará en cuenta que el código tenga un footer con la fecha del día que se ejecute, esto debe ser incluido como un archivo externo.

7. Se tomará en cuenta que utilice funciones de validación de datos (preg\_match, filter\_var, htmlspecialchars).

<https://www.php.net/manual/es/function.preg-match.php>

<https://www.php.net/manual/es/function.htmlspecialchars.php>

<https://www.php.net/manual/es/function.filter-var.php>



## 8. Utilizar formularios.

## 9. Utilizar clases de utilidades con métodos estáticos.

```
    public static function validarEnLista($valor, $lista_permitida)
    {
        return in_array($valor, $lista_permitida);
    }

    public static function nvl(& $var) {
        /* if $var is undefined, return $default, otherwise return $var
         * $default="";
         */
        return isset($var) ? $var : $default;
    }

    /**
     * Validar token CSRF
     */
    public static function validarCSRF() {
```

## 9. Realizar un menú que permita seleccionar el problema.

## 10. Utilizar un enlace para volver al menú principal. Que ese enlace esté en una función, pasar el parámetro URL. (**Navegación**)

## Enunciado de los problemas

<p><b>Calcular la media, desviación estándar, el número min y el máx de los 5 primeros números positivos introducidos a partir de un formulario.</b></p>	<ul style="list-style-type: none"> <li>• Imprimir los N – primeros múltiplos de 4, donde N es un valor introducido por teclado.  <math>(4*1=4;</math>  <math>4*2=;4*3=)</math>...Desbordamiento?</li> </ul>	<p>Calcular la suma de los números del 1 al 1,000.      500500 problema #2</p>
<p><b>Problema #1</b></p>	<p><b>Problema #3</b></p>	<p><b>Problema #2</b></p>
<p>Se desea calcular independientemente la suma de los número pares e impares comprendidos entre 1 y 200.</p>	<p>Leer la edad de 5 personas y clasificar cada una en una categoría: niño (0-12), adolescente (13-17), adulto (18-64), adulto mayor (65+).      Generar Estadísticas si se repiten las edades. Integrar gráficas.</p>	<p>En un hospital existen tres áreas: Ginecología, Pediatría y Traumatología. El presupuesto anual del hospital se reparte conforme a la siguiente tabla      Ver Referencia página #7      Integrar Gráficas.</p>
<p><b>Problema #4</b></p>	<p><b>Problema #5</b></p>	<p><b>Problema #6</b></p>
<p><b>Calculadora de Datos Estadísticos</b>      Pedir la cantidad de notas que desea ingresar el usuario. Luego pedir esas notas y calcular el promedio, la desviación estándar, la nota, mínima y la máxima. Usar foreach (o un ciclo que recorra una colección).</p>	<p><b>Estación del Año</b>      Al ingresar la fecha, devolver la estación de año de acuerdo con la siguiente tabla (ver imagen de la estación del Año)      Ver Referencia página #7      Datos prueba.</p>	<p>Solicitar un número (1 al 9)      Generar o imprimir las 15 primeras potencias del número ( 4 elevado a la 1, 4 elevado a la dos, ....  <b>Problema #9</b></p>
<p><b>Problema #7</b></p>	<p><b>Problema #8</b></p>	<p><b>Problema #9</b></p>

Utilice un arreglo bidimensional para resolver el siguiente problema.

Una empresa tiene cuatro vendedores (1 al 4) que venden cinco productos diferentes (1 al 5). Una vez al día, cada empleado pasa en una nota para cada tipo diferente de producto vendido. Cada hoja contiene lo siguiente:

- a) El número del vendedor
- b) El número de producto
- c) El valor total en dólares de ese producto vendido ese día

Así, cada vendedor pasa entre 0 y 5 notas de ventas por día. Suponga que está disponible la información sobre todas las notas del mes pasado. Escriba un programa que lea toda esa información para las ventas del último mes (los datos de un vendedor a la vez (Nombre, Apellido, Número de Vendedor) y que resuma las ventas totales por vendedor, por producto.

Todos los totales deben guardarse en el arreglo bidimensional ventas. Después de procesar toda la información del mes pasado, muestre los resultados en formato tabular, en donde cada columna representa a un vendedor específico y cada fila representa un producto. Saque el total de cada fila para obtener las ventas totales de cada producto durante el último mes. Saque el total de cada columna para obtener las ventas totales por cada vendedor durante el último mes. Su impresión tabular debe incluir estos totales cruzados a la derecha de las filas totalizadas y en la parte inferior de las columnas totalizadas.

### Problema #10

ESTACIÓN	TIEMPO
Verano	Del 21 de diciembre al 20 de marzo
Otoño	Del 21 de marzo al 21 de junio
Invierno	Del 22 de junio al 22 de Setiembre
Primavera	Del 23 de setiembre al 20 de diciembre

QuestionPro

Fórmula para calcular la desviación estándar

$$S = \sqrt{\frac{\sum (x - \bar{x})^2}{n - 1}}$$

Área	Porcentaje del presupuesto
Ginecología	40%
Traumatología	35%
Pediatría	25%

En un hospital existen tres áreas: Ginecología, Pediatría y Traumatología. El presupuesto anual del hospital se reparte conforme a la siguiente tabla.

## Definición de Potencia

Una potencia es una operación matemática que consiste en multiplicar un número por sí mismo varias veces. El número que se multiplica se denomina base, y el número de veces que se multiplica se denomina exponente.

## Fórmula de la Potencia

La fórmula general de una potencia se expresa como  $a^n$ , donde  $a$  es la base y  $n$  es el exponente. Esto significa que la base  $a$  se multiplica por sí misma  $n$  veces. Por ejemplo,  $a^3 = a \times a \times a$ .

## Primeras 15 Potencias del Número 4

Salidas Esperadas:

## ¿Qué estación es?



**Mostrar**

Fecha ingresada: **25-09**

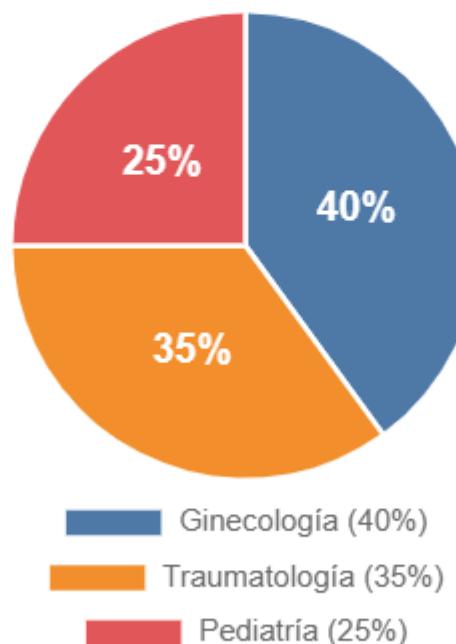
La estación es: **Primavera**



## Resultados del Presupuesto

- 妇科 (Ginecología): 40%: \$8,000.00
- 创伤学 (Traumatología): 35%: \$7,000.00
- 儿科 (Pediatría): 25%: \$5,000.00

Distribución del presupuesto: \$20000



## Apéndice A. Secuencias de comandos entre sitios (XSS)

Las vulnerabilidades de scripts entre sitios (XSS) se producen cuando un atacante puede injectar scripts maliciosos en una aplicación web. **Esto puede ocurrir cuando la entrada del usuario no se depura ni valida adecuadamente.** Los ataques XSS pueden provocar el robo de datos, el robo de cookies o incluso la vulneración total del sistema.

Para evitar ataques XSS en aplicaciones PHP, debes:

- Utilice la función htmlspecialchars() de PHP para codificar la entrada del usuario y evitar la ejecución de scripts.

A continuación, se muestra un ejemplo de cómo utilizar la función htmlspecialchars() de PHP para evitar ataques XSS:

```
echo 'Hola ' . htmlspecialchars( $_GET[ "nombre" ] ) . '!' ;
```

Referencia: <https://medium.com/@khalidzeiter/php-owasp-top-10-essential-steps-for-keeping-your-web-application-safe-and-secure-72b5e7e55523>

## Apéndice B. Top 10 Web Application Security Risks

<https://owasp.org/www-project-top-ten/>

