

<b>Instrucciones</b>	<b>Consejos</b>
<ul style="list-style-type: none"> <li>- <b>Crear una carpeta</b> que se llamará “Apellidos_Nombre”, donde <b>guardaréis las soluciones de los problemas</b></li> <li>- Para cada problema crearéis una carpeta que llamaréis “Problema1”, “Problema2”, etc, donde guardaréis <b>las soluciones</b>.</li> <li>- Una vez hayáis finalizado debéis subir el examen al aula virtual comprimido en un fichero con el formato “Apellidos_Nombre”.</li> <li>- El examen puntúa sobre 10, selecciona los ejercicios como quieras.</li> <li>- Si se utiliza Javascript no será tenido en cuenta.</li> </ul>	<ul style="list-style-type: none"> <li>- Guardar cada poco el trabajo que estéis realizando.</li> <li>- Mirar la puntuación de los problemas y centraros primero en el aprobado. Luego incluís mejoras.</li> <li>- El programa es personal e intransferible, por lo que no deberíais tener que mirar para nada el trabajo del compañero.</li> <li>- Escoger en primer lugar los problemas que tienen fácil solución, asegurad el aprobado lo primero.</li> <li>- Recordad que podéis hacer uso de código que ya tenéis creado</li> </ul>

Antes de comenzar, leed todos los problemas para ver cual es cual es vuestra mejor estrategia de solución del examen.

## Problema 1 – Secuencia de números – 2 Ptos

El nombre del fichero con el que arranca el programa será **index.php**. En este fichero incluiréis al comienzo un comentario en el que indicaréis para que sirven el resto de ficheros utilizados en la solución (en caso necesario).

Se debe crear un formulario que pida, mediante controles de texto, los siguientes valores numéricos.

- Los valores *Límite superior* y *Límite inferior* indican los límites de un rango de números. Se debe chequear que sean valores numéricos mayores que 0 y que el límite superior sea mayor o igual que el límite inferior.
- *Número* indica un valor numérico que debe estar entre Límite inferior y Límite superior, ambos inclusive.
- *Repeticiones* indica el número de veces que debe darse número.

Cuando el usuario pulse el botón *Generar números* se validará la información del formulario y, en caso de que ésta sea correcta, se generarán números aleatorios comprendidos entre Límite inferior y Límite superior hasta que se produzcan las repeticiones indicadas del Número introducido. Estas repeticiones se mostrarán en color rojo.

Puntuación (sobre 10):

- Filtrar datos entrada: 2,5 pts.
- Entrada con memoria: 2 pts.
- Presentación y retroalimentación al usuario: 2 pts.
- Programa completo: 3,5 pts.

### Secuencia de números

Límite inferior

Límite superior

Número

Repeticiones

**Generar números**

### Secuencia de números

Límite inferior

Límite superior

¡El límite superior no puede ser mas bajo que el inferior!

Número

¡El número debe ser un valor numérico!

¡El número no está entre el valor inferior y el superior!

Repeticiones

**Generar números**

### Secuencia de números

Límite inferior

10

Límite superior

15

Número

10

Repeticiones

3

Generar números

Secuencia de números generada:

13, 11, **10**, 11, **10**, 15, 15, 12, 11,  
12, 11, **10**

## Problema 2 – Cuestionario (4 pts)

El nombre del fichero con el que arranca el programa será **index.php**. En este fichero incluiréis al comienzo un comentario en el que indicaréis para que sirven el resto de ficheros utilizados en la solución.

Realice una página que nos permita hacer cuestionarios a los usuarios. La página mostrará un test de evaluación con un número variable de preguntas sobre alguna materia. El cuestionario se construirá a partir de una matriz que contendrá las preguntas, las posibles opciones y las respuestas. Una vez se muestre el test al usuario, este podrá marcar las respuestas que considere correctas y pulsará el botón Enviar para ver el resultado de su intento.

Podemos encontrar un ejemplo de la matriz cuestionario en el fichero *cuestionario.php*. Para este examen podéis incluir dicho fichero en el código y así tendréis un cuestionario de ejemplo con el que probar vuestro programa. En una situación real usaríamos ficheros json o xml.

Puntuación (sobre 10):

- Mostrar el cuestionario: 3 pts.
- Mostrar aciertos, fallos y respuestas en blanco del cuestionario: 3 pts.
- Presentación y retroalimentación al usuario: 2 pts.
- Programa completo: 2 pts.

### Cuestionario:

Año descubrimiento América

- ☐ 1492  
☐ 1692  
☐ 1992

Año fin 1ª Guerra mundial

- ☐ 1945  
☐ 1936  
☐ 1918

Año inicio 2ª Guerra mundial

- ☐ 1945  
☐ 1939  
☐ 1918

País más poblado del mundo

- ☐ India  
☐ China  
☐ EEUU

Enviar

Reiniciar

## Cuestionario:

Año descubrimiento América

- ☐ 1492  
☒ 1692  
☐ 1992

Año fin 1ª Guerra mundial

- ☐ 1945  
☐ 1936  
☒ 1918

Año inicio 2ª Guerra mundial

- ☐ 1945  
☒ 1939  
☐ 1918

País más poblado del mundo

- ☐ India  
☒ China  
☐ EEUU

Enviar

Reiniciar

## Resultado cuestionario:

Aciertos: 2

Fallos: 1

En blanco: 1

Al mostrar el resultado del cuestionario las soluciones a las preguntas siempre se muestran en verde y tus respuestas incorrectas se muestran en rojo.

## Problema 3 – Médicos (7,5 pts)

El nombre del fichero con el que arranca el programa será **index.php**. En este fichero incluiréis al comienzo un comentario en el que indicaréis para que sirven el resto de ficheros utilizados en la solución.

Se desea construir una aplicación que nos permita realizar un CRUD de una tabla de médicos. El *script* de creación de la tabla *medicos* se encuentra en el fichero *sc\_exaEva1A.sql* suministrado junto con el examen. Al entrar al programa lo primero que se mostrará es una página de consulta de médicos desde la que se podrá acceder a cada una de las funcionalidades de la aplicación. El estilo a seguir en el diseño de la página es libre y podéis utilizar código ya existente.

**Listado de Médicos** [+Nuevo médico](#) Registros por página ▾

Búsqueda por:

Nombre	Apellidos	Nivel	Salario	Acción
Ana	Ríos	1	100000	<a href="#">✎</a> <a href="#">✖</a>
Cinta	García	2	100000	<a href="#">✎</a> <a href="#">✖</a>
David	Sánchez	3	150000	<a href="#">✎</a> <a href="#">✖</a>
Elena	Sánchez	2	100000	<a href="#">✎</a> <a href="#">✖</a>
Esteban	Mesa	2	100000	<a href="#">✎</a> <a href="#">✖</a>

« 1 2 3 »

En el alta y modificación de los datos de un médico se debe tener en cuenta que los campos nombre, apellidos y nivel son obligatorios. Además, el nivel del médico debe ser un número entero mayor o igual que 0 y el salario un número real mayor que 0.

Puntuación (sobre 10):

- Mostrar lista de médicos (sin paginar): 1,5 ptos.
- Mostrar lista de médicos (paginado): +1 pto.
- Filtrar la lista de médicos por campos de búsqueda: +1,5 ptos.
- Modificar información médico: 2 ptos.
- Borrar médico: 1,5 ptos.
- Dar de alta nuevo médico: 1,5 ptos.
- Programa completo: 1 pto.

## Problema 4 – 3 ptos.

El nombre del fichero con el que arranca el programa será **index.php**. En este fichero incluiréis al comienzo un comentario en el que indicaréis para que sirven el resto de ficheros utilizados en la solución.

Se trata de construir un formulario de entrada de datos que valide la información introducida según una serie de criterios. Al pulsar el botón Enviar, si la información es correcta, se mostrarán todos los datos introducidos, si la información no cumple los criterios se mostrarán advertencias avisando de los errores encontrados en la validación.

El formulario tendrá un diseño similar a este:

### Rellene la encuesta

Nombre	<input type="text" value="Nombre usuario"/>
Fecha	<input type="text" value="dd/mm/aaaa"/>
Correo electrónico	<input type="text" value="email"/>
Ciudad	<input type="text" value="Escoja ciudad..."/>
Intereses	<input type="checkbox"/> Interés 1 <input type="checkbox"/> Interés 2 <input type="checkbox"/> Interés 3
¿Aceptas términos?	<input checked="" type="radio"/> Sí <input type="radio"/> No
<input type="button" value="Enviar"/>	

Los criterios de validación son los siguientes:

- El campo **nombre** es obligatorio y de longitud al menos 2 caracteres.
- El campo **fecha** es obligatorio y sigue el formato *dd/mm/aaaa*
- El campo **correo** electrónico es opcional y sigue el formato de email
- La **ciudad** se escoge de entre las opciones que aparecen en un desplegable cuyos valores han sido obtenidos de una tabla de base de datos denominada *provincias*. Esta tabla se crea al ejecutar el script *sc\_exaEva1A.sql* suministrado junto con el examen. El usuario debe seleccionar una ciudad.
- Es obligatorio seleccionar **al menos un interés**.

## Rellene la encuesta

Nombre

¡El nombre no puede estar vacío!

Fecha

Correo electrónico

Ciudad

Intereses ☒ Interés 1 ☒ Interés 2 ☐ Interés 3

¿Aceptas términos? ☒ Sí ☐ No

## Rellene la encuesta

Nombre

Fecha

Correo electrónico

Ciudad

Intereses ☒ Interés 1 ☒ Interés 2 ☐ Interés 3

¿Aceptas términos? ☒ Sí ☐ No

### Datos introducidos:

Nombre: Pedro

Fecha: 15/04/2020

Correo: asi@si.es

Ciudad: 3

Intereses: Interés 1 Interés 2

¿Aceptas?: Sí

Puntuación (sobre 10):

- Filtrar y validar los datos de entrada según criterios: 2,5 pts.
- Entrada con memoria: 2 pts
- Cargar información de las ciudades: 2,5 pts.
- Presentación y retroalimentación al usuario: 1 pto.
- Programa completo: 2 pts.