

## Escuela de Ingeniería

Segundo Parcial de: Programación I

Código de materia:

Fecha: Julio 2019

Grupo: EXTRA

Hoja 1 de 3

Duración:

3 horas

Con material: No

Puntaje máximo:

35 puntos

### Ejercicio 1) Fábrica de bolsas (máximo 21 puntos)

Debido a las nuevas reglamentaciones sobre el uso de bolsas, la empresa “Bolsas de Papel” necesita una nueva página web donde ofrecer sus productos.

#### Diseño HTML+CSS

**1a) Se pide (máximo 7 puntos):** Completar el código HTML de la única página index.html y la hoja de estilo (estilos.css). El código HTML deberá ir dentro de una etiqueta <article>. El resto de la página NO se solicita y se asume disponible.

Archivo **index.html**

<article id="idArticulo">

**CODIGO A COMPLETAR**

</article>

Archivo **estilos.css**

**CODIGO A COMPLETAR**

La parte de página solicitada debe ser similar a la que se muestra:

## Sobre nosotros

Somos una empresa orientada a la fabricación de bolsas

Tenemos bolsas de:

1. Papel de lujo
2. Papel con asa plana
3. Papel para botella

## Detalle de las bolsas

Las bolsas se fabrican en papel resistente

Tamaños:

Descripción: Contiene un título (“Sobre nosotros”), dos párrafos (“Somos una empresa...” y “Tenemos bolsas...”), una lista numerada con el detalle de 3 tipos de bolsas (1. Papel de lujo, 2. Papel con asa plana, 3. Papel para botella), otro título (“Detalle de las bolsas”), un párrafo (“Las bolsas se ...”) y un combo de “Tamaños” con las opciones: “Bolsa chica”, “Bolsa mediana” y “Bolsa grande”.

El formato de la etiqueta article debe ser: color de fondo #F3F3E5 y con separación a la izquierda de 5%.

## Escuela de Ingeniería

**Segundo Parcial de: Programación I**

**Código de materia:**

**Fecha: Julio 2019**

**Grupo: EXTRA**

**Hoja 2 de 3**

### Código JS y HTML

En la página index.html se tiene un formulario para realizar consultas sobre los bolsos de hasta cierto precio y que sean para un uso determinado. Se ingresará el tope de precio y la palabra sobre su finalidad. Se tiene:

```
<h2>Consulta de precios</h2>
<form id="idFormC">
  <label for="idFinalidad">Finalidad</label>
  <input type="text" required placeholder="Finalidad" maxlength="30" id="idFinalidad">
  <label for="idTope">Tope de precio </label>
  <input type="number" required placeholder="Tope" min="0" max="4000" id="idTope">
  <input type="button" value="Consultar" id="idBotonC">
</form>
<ul id="idLista">
</ul>
```

Se tienen además las clases:

```
class Bolsa{
  constructor(descripcion, precio){
    this.precio=precio;
    this.descripcion= descripcion;
  }
}
class Empresa{
  constructor(){
    this.listaBolsas=[];
  }
  agregarBolsa(unaBolsa){
    this.listaBolsas.push(unaBolsa);
  }
  darBolsasTipoPrecio(palabra,hasta){
    // a completar
  }
}
```

**1b) Se pide: (máximo 6 puntos):** Completar en la clase Empresa el método darBolsasTipoPrecio(palabra, hasta) que recibe una palabra y un precio tope y debe retornar un array indexado con las bolsos que contengan esa palabra en su descripción y no superen el tope. Debe estar ordenada por precio creciente.

**1c) Se pide: (máximo 8 puntos):** Se tiene definida una instancia de la Empresa **let empresa = new Empresa();** y se asoció este evento: **document.getElementById("idBotonC").addEventListener("click", cargaLista);**. Implementar la función cargaLista que, si el formulario es válido, toma la palabra y precio de pantalla, invoca al método "darBolsasTipoPrecio" con esos valores y carga el resultado en la lista de id: "idLista", de a una bolsa por línea. Incluir además el método y código necesario para que de cada bolsa se muestre cada letra de su descripción en mayúscula separada por guiones y el precio. Si no hay datos, indicarlo.

Ejemplo: si la descripción es "Comida" y el precio es 400 se debe mostrar:

- C-O-M-I-D-A \$400

**Escuela de Ingeniería**

**Segundo Parcial de: Programación I**

**Código de materia:**

**Fecha: Julio 2019**

**Grupo: EXTRA**

**Hoja 3 de 3**

**Ejercicio 2: Arrays (máximo 14 puntos)**

**2a) (máximo 7 puntos):** Implementar en JS la función `pacientesMuchasConsultas` que recibe un array con los números de paciente que consultó en un cierto período y retorna un nuevo array indexado con los números de pacientes (sin repeticiones) que hicieron más de una cantidad dada de consultas. Si hubiera más de 20 pacientes en estas condiciones, retornar 20 de ellos.

**Encabezado: `function pacientesMuchasConsultas(datos, cantidad)`**

Ejemplo::

```
var cons = [4, 1, 5, 1, 5, 3, 6, 89, 89, 4, 7, 2, 4, 7, 89, 4, 4, 3];
```

```
alert(pacientesMuchasConsultas(cons,2));
```

saldría un array con 4 y 89

**2b) Se pide: (máximo 7 puntos):** Implementar en JS la función `palindromoMasLargo(letras)` que recibe un array. En cada posición hay una letra minúscula. Un palíndromo es una palabra que se lee igual de izquierda a derecha que al revés. Ejemplos: NEUQUEN, ANA. Se debe retornar el largo del palíndromo más largo (comenzando desde la primera letra).

Ejemplo:

si se invoca: `palindromoMasLargo(["a","n","a","n","a","s"])` debe retornar 5 (correspondiente a "anana")

si se invoca: `palindromoMasLargo(["a","n","b","r","a","s"])` debe retornar 1

**Encabezado: `function palindromoMasLargo(letras){`**