

Escuela de Ingeniería

2do Parcial de: Programación I Presencial

Código de materia:

Fecha: Julio 2024

Grupo: MATUTINO VESPERTINO

Hoja 1 de 2

v5

Duración:	2 horas	Con material:	NO	Puntaje máximo:	40 puntos
-----------	---------	---------------	----	-----------------	-----------

1) (8 puntos)

Dado un array con números del 1 al 300 (puede haber repetidos), se considera que un número es "pico" cuando el número ubicado antes y el número ubicado después son ambos estrictamente menores.

Ejemplo: [111, **234**, 200, 110, 143, **159**, 120, **234**, 8, 45]

Los picos son: 234, 159 y 234.

Implementar una función en JS que recibe el array y retorna la cantidad de picos diferentes.

firma: **function cantidadPicosDiferentes(datos)**

Para el ejemplo, retorna 2.

2) (12 puntos)

Una frase es **similar** a otra si, ignorando mayúsculas y minúsculas, empiezan con la misma letra y terminan con los mismos 3 caracteres. Las frases se asumen correctas.

Ejemplo: la frase: "Historia parcial JS" es similar a: "hitos del lenguaje Js"

Se tienen dos arrays que contienen cada uno una frase en cada posición. Implementar en JS una función que recibe los dos arrays y retorna cuántas frases del primer array tienen una o más frases similares en el segundo array.

firma: **function cantidadConSimilares(datos1, datos2)**

Ejemplo:

datos1: ["Historia de JS", "Procesar archivos", "Explicar como es un objeto"]

datos2: ["Ver historia de JS", "Mas de multiplicar", "hitos del lenguaje JS", "pedir muchos datos", "historia del lenguaje Js"]

Retorna 1

3) La competencia de Ironman consiste en nadar 3.8 km, andar en bicicleta 180 km y correr 42.2 km. En el sistema se registran distintos momentos de la competencia: competidor, disciplina y distancia. Los registros no están ordenados. Se tiene estas clases:

```
class Competidor {
  constructor(numero, nombre){
    this.numero = numero;
    this.nombre = nombre;
    this.listaRegistros = []; //lista de objetos RegistroDistancia
  }
}
class RegistroDistancia{
  constructor(competidor, disciplina, distancia){
    this.competidor=competidor; //Objeto Competidor
    this.disciplina = disciplina; //1-Nadar, 2-Bicicleta, 3-Correr
    this.distancia = distancia;
  }
}
```

```
class Sistema{
  constructor(){
    this.listaCompetidores = [];
    this.listaRegistrosDistancia =[];
  }
  // pedido 3a
  competidoresEnLasTres(){
    // codigo a completar
  }
}
```

3a) (máximo 9 puntos) En la clase Sistema, implementar el método **competidoresEnLasTres()**, que retorna un array con los competidores que tiene registro/s de distancia en cada una de las 3 pruebas.

## Escuela de Ingeniería

**2do Parcial de: Programación I Presencial**

**Código de materia:**

**Fecha: Julio 2024**

**Grupo: MATUTINO VESPERTINO**

**Hoja 2 de 2**

**v5**

**3b) (máximo 2 puntos)** Asumir disponible la página HTML (vista parcial) a continuación y el CSS:

```
<h1>Datos de Natación</h1>
<input type="button" id="idBoton">
<table id="idTabla" class="p">
</ table >
```

Anotar el HTML para incluir un párrafo de id "idParrafo" luego de la tabla.

Agregar el código CSS para todas las tablas de la página tengan color de fondo amarillo.

**3c) (máximo 9 puntos)** Al presionar el botón, se carga en la tabla todos los registros de distancia correspondientes a natación, e incluir el nombre del competidor en cada uno. En caso de no existir registros a mostrar, se deberá cargar en el párrafo "idParrafo" el texto: "Sin datos". Si se mostraron datos en la tabla, dejar el párrafo vacío.

Ejemplo de la tabla:

3.8 Ana
3.4 Luis
3.7 Pedro
3.7 Ana

En el archivo fuentes.js se tiene:

```
window.addEventListener("load", inicio);
let sistema = new Sistema(); // contiene la instancia de Sistema
function inicio(){
    document.getElementById("idBoton").addEventListener("click", proceso);
}
// parte 3c)
function proceso(){
    // codigo a completar
}
```