



# Laboratorio 2

## Análisis

Ricardo Arturo Godínez Sánchez #23247

### ▼ Requisitos funcionales

- Asignar cursos al horario del salón los días y horarios que el curso deba impartirse.
- Visualizar la información del curso que está en un horario determinado, que elija ver el usuario.
- Eliminar un curso asignado de un período de tiempo del calendario en un día específico.
- Mostrar un mensaje cuando la cantidad de estudiantes inscritos sea mayor que la del laboratorio.
- Mostrar un mensaje de error cuando la cantidad de estudiantes inscritos en el curso sea más del doble de la capacidad del laboratorio. Por cuestiones de acreditación no se permiten más de 2 estudiantes por computadora por lo que no se puede asignar el curso en ningún horario del lab.

- Cambiar el curso que se impartirá de horario, siempre verificando que el laboratorio esté libre en el nuevo horario que se quiera impartir el curso.
- Poner disponible el laboratorio en todos los horarios para el nuevo semestre.
- Mostrar el profesor encargado en un día y horario determinado.
- Mostrar los días en los cuales un profesor está al frente del laboratorio, tenga en cuenta que un profesor puede dar más de un curso en diferentes días y horarios, así que al mostrar los datos del profesor debe incluir también los datos del curso y del horario. El usuario debe ingresar el nombre del profesor.
- Mostrar cuántas veces un mismo profesor está al frente del laboratorio y el porcentaje de responsabilidad que tiene con respecto al laboratorio, es decir si un profesor está 10 veces de los n posibles horarios del laboratorio tiene más responsabilidad que uno que está 3 veces en la semana.

## ▼ Clases

### *Curso*

#### Atributos

- El código del curso: codigo: String, private
- el nombre del curso: nombre: String, private
- la cantidad de períodos: periodos: int, private
- los días de la semana que se imparte el curso: diasSemana: String, private
- el horario en que se imparte el curso: horario:int, private
- la cantidad de estudiantes asignados: cantEstudiantes: int, private
- el profesor que imparte el curso

### *Profesor*

El profesor es el responsable de la disciplina y el cuidado de los insumos dentro del laboratorio, por lo que es necesario conocer su información

#### Atributos

- El nombre completo: nombre: String, private
- El carné: carne: int, private
- El correo electrónico: correo: String, private
- El teléfono: telefono: int, private

## *Salon*

### Atributos

- La matriz en donde se guarda el horario. Curso [][]

### Métodos

- **crearProfesor** : Se crea un profesor que se guardará en una lista
- **:asignarCurso** : Asigna un curso al horario del laboratorio.
- **visualizarCurso** : Muestra la información del curso en un horario y día específicos.
- **eliminarCurso** : Elimina un curso de un horario y día específicos.
- **cambiarHorarioCurso** : Cambia el horario de un curso.
- **mostrarProfesor** : Muestra el profesor encargado en un día y horario específicos.
- **diasProfesor** : Muestra los días en que un profesor está al frente del laboratorio.
- **responsabilidadProfesor** : Calcula el porcentaje de responsabilidad de un profesor con respecto al laboratorio.

## *GestionLab*

La clase GestionLab se tendrá el horario y la capacidad de los estudiantes que caben en el.

### Atributos

- horario: Matriz que nos ayudará a ordenar los espacios.

### Métodos

- `crearProfesor` : Deberá recibir todos los atributos de la clase profesor
- `asignarCurso` : Deberá recibir todos los atributos de la clase cursos
- `visualizarCurso` : Deberá recibir los datos que ingresará el usuario, y de la matriz y lista los mostrará en pantalla
- `eliminarCurso` : Deberá recibir los datos que ingresará el usuario y elimina un curso de un horario y día específicos.
- `cambiarHorarioCurso` : Deberá recibir los datos que ingresará el usuario para cambiar el horario de un curso.
- `mostrarProfesor` : Deberá recibir los datos que ingresará el usuario para mostrar el profesor encargado en un día y horario específicos.
- `diasProfesor` : Deberá recibir el nombre de el profesor
- `responsabilidadProfesor` : Deberá recibir el nombre de el profesor.

