



**Recuperatorio
Primer Parcial**

***Programación I – Laboratorio I.
Tecnicatura Superior en Programación.
UTN-FRA***

Autor: Tomas Agustin Gauna

División: B

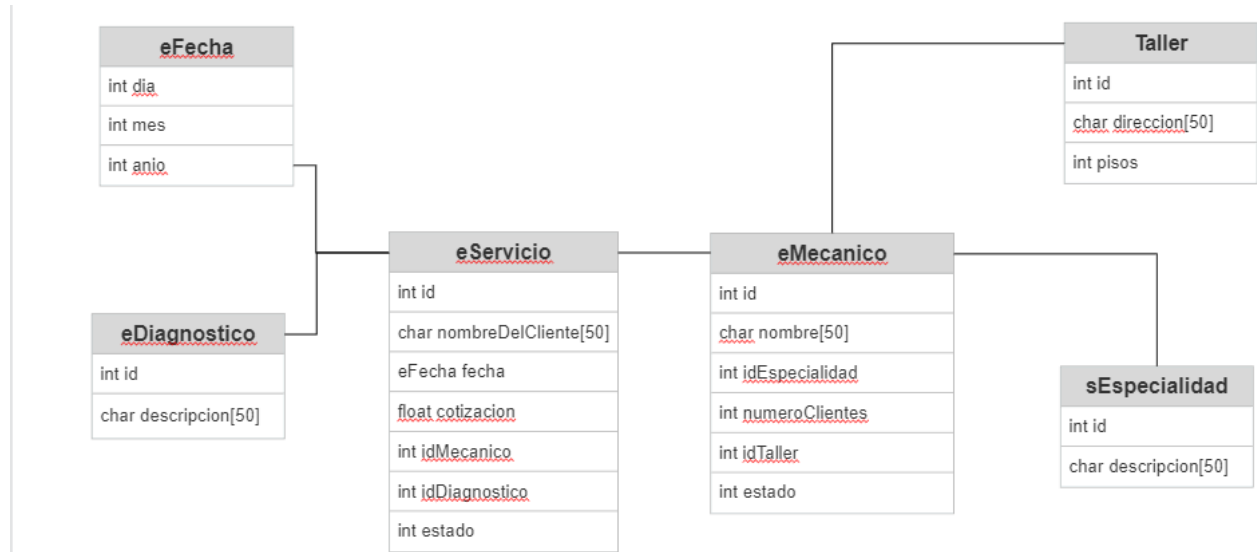
Dni: 45.751.587

Docentes: Scarafilo, Taboada, Lucchetta, Fernandez.

Nueva estructura agregada: "Taller".

La estructura agregada "Taller" contiene un identificador (id), un domicilio (dirección), y una cantidad de pisos (pisos). Está directamente relacionada con la estructura "eMecanico", ya que va a ser el mecánico el que tenga asignado un taller por defecto, sería el lugar donde trabaja.

Diagrama Entidad Relacion (ERD)



Prototipo del informe anexo.

@brief Ordena la lista de servicios alfabeticamente por direccion de taller del mecanico

@param listaServicios array de lista de servicios cargada por el usuario o hardcodeada

@param sizeServicio largo de la lista de servicios(T)

@param listaMecanicos array de lista de mecanicos hardcodeada

@param sizeMecanico largo de la lista de mecanicos (M)

@param listaTaller array de la lista de talleres hardcodeada

@param sizeTaller largo de la lista de talleres (L)

@param listaEspecialidad array de la lista de especialidades hardcodeada

@param sizeEspecialidad largo de la lista de especialidades (E)

```

void OrdenarServiciosPorDireccionTaller(eServicio listaServicios[], int sizeServicio,
eMecanico listaMecanicos[], int sizeMecanico, Taller listaTaller[], int sizeTaller,
sEspecialidad listaEspecialidad[], int sizeEspecialidad);
    
```

Prototipo de cada función del núcleo del programa.

- **Biblioteca "servicio.h":**

```
/// @brief Busca la primer posicion libre dentro del array
///
/// @param listaServicios array de servicios
/// @param sizeServicios logintud del array de servicios
/// @return int posicion libre encontrada (-1)si no encuentra posicion libre
int BuscarEspacioLibre(eServicio listaServicios[], int sizeServicios);
```

```
/// @brief Calcula el id automaticamente
///
/// @return el id anterior incrementado en 1
int CalcularIdServicio();
```

```
void PedirNombre(eServicio unServicio);
int PedirDia();
int PedirMes();
int PedirAnio();
```

```
/// @brief Pide los datos de un servicio
///
/// @param listaServicios array de servicios
/// @param sizeServicios logintud del array de servicios
/// @return el servicio completado
eServicio PedirUnServicio();
```

```
/// @brief Carga una lista de servicios
///
/// @param listaServicios array de servicios
/// @param sizeServicios longitud array de servicios
/// @return (-1) si no hay espacio (1)si esta ok
int CargarListaDeServicios(eServicio listaServicios[], int sizeServicios);
```

```
/// @brief Busca un id ingresado por parametro
///
/// @param listaServicios array de servicios
/// @param sizeServicios longitud del array de servicios
/// @param id int a buscar en el array
/// @return (-1) si no se encontro (i)posicion en donde se encuentra el id
int BuscarId(eServicio listaServicios[], int sizeServicios, int id);
```

```
/// @brief Muestra los datos de un servicio
///
/// @param unServicio servicio a mostrar
void MostrarUnServicio(eServicio unServicio);
```

```
/// @brief Muestra los datos de toda la lista de servicios
///
/// @param listaServicios array de lista de servicios cargada por el usuario o hardcodeada
/// @param sizeServicio largo de la lista de servicios(T)
void MostrarListaDeServicios(eServicio listaServicios[], int sizeServicios);
```

```
/// @brief Muestra los datos de la lista de servicios que no han sido diagnosticados
///
/// @param listaServicios array de lista de servicios cargada por el usuario o hardcodeada
/// @param sizeServicios largo de la lista de servicios(T)
void MostrarServiciosNoDiagnosticados(eServicio listaServicios[], int sizeServicios);
```

```
/// @brief Modifica el nombre, la fecha o el mecanico del servicio elegido por el usuario
///
/// @param listaServicios listaServicios array de lista de servicios cargada por el usuario o
hardcodeada
/// @param sizeServicios largo de la lista de servicios(T)
/// @param listaMecanicos array de lista de mecanicos hardcodeada
/// @param sizeMecanico largo de la lista de mecanicos (M)
/// @param listaEspecialidades array de la lista de especialidades hardcodeada
/// @param sizeEspecialidad largo de la lista de especialidades (E)
/// @return (-1) si no pudo modificar y 1 si esta ok
int ModificarServicio(eServicio listaServicios[], int sizeServicios, eMecanico listaMecanicos[],
int sizeMecanico, sEspecialidad listaEspecialidades[], int sizeEspecialidad);
```

```
/// @brief Modifica el servicio segun la opcion elegida por el usuario
///
/// @param opcionElegida opcion que eligio el usuario para modificar
/// @param listaServicios array de lista de servicios cargada por el usuario o hardcodeada
/// @param index posicion del array en la que se va a modificar el dato
/// @param listaMecanicos array de lista de mecanicos hardcodeada
/// @param sizeMecanico largo de la lista de mecanicos (M)
/// @param listaEspecialidades array de la lista de especialidades hardcodeada
/// @param sizeEspecialidad largo de la lista de especialidades (E)
/// @return (-1) si hubo algun error y 1 si esta ok
int MostrarMenuModificacion(int opcionElegida, eServicio listaServicios[], int index, eMecanico
listaMecanicos[], int sizeMecanico, sEspecialidad listaEspecialidades[], int sizeEspecialidad);
```

```
/// @brief Da de baja un servicio de la lista de servicios
///
/// @param listaServicios array de lista de servicios cargada por el usuario o hardcodeada
/// @param sizeServicios largo de la lista de servicios(T)
/// @param listaMecanicos array de lista de mecanicos hardcodeada
/// @param sizeMecanico largo de la lista de mecanicos (M)
/// @return (-1) si hubo algun error y 1 si elimino
int DarDeBaja(eServicio listaServicios[], int sizeServicios, eMecanico listaMecanicos[], int
sizeMecanicos);
```

```
/// @brief Diagnostica un servicio de la lista de servicios
///
/// @param listaServicios array de lista de servicios cargada por el usuario o hardcodeada
/// @param sizeServicios largo de la lista de servicios(T)
/// @param listaMecanicos array de lista de mecanicos hardcodeada
/// @param sizeMecanico largo de la lista de mecanicos (M)
/// @param listaDiagnosticos array de la lista de diagnosticos harcodeada
/// @param sizeDiagnostico largo de la lista de diagnosticos (D)
/// @param listaEspecialidades array de la lista de especialidades hardcodeada
/// @param sizeEspecialidad largo de la lista de especialidades (E)
/// @return (-1) si hubo algun error y 1 si esta ok
```

```
int DiagnosticarServicio(eServicio listaServicios[], int sizeServicios, eMecanico
listaMecanicos[], int sizeMecanico, eDiagnostico listaDiagnosticos[], int sizeDiagnostico,
sEspecialidad listaEspecialidades[], int sizeEspecialidad);
```

```
/// @brief Ordena la lista por fecha y la muestra
```

```
///
```

```
/// @param listaServicios array de lista de servicios cargada por el usuario o hardcodeada
```

```
/// @param sizeServicios largo de la lista de servicios(T)
```

```
void MostrarListaOrdenadaFecha(eServicio listaServicios[], int sizeServicios);
```

```
/// @brief Muestra la lista de servicios diagnosticados por 'Cambio de aceite', segun
determinada fecha
```

```
///
```

```
/// @param listaServicios array de lista de servicios cargada por el usuario o hardcodeada
```

```
/// @param sizeServicios largo de la lista de servicios(T)
```

```
/// @param listaDiagnosticos array de la lista de diagnosticos harcodeada
```

```
/// @param sizeDiagnostico largo de la lista de diagnosticos (D)
```

```
void MostrarCotizacionSegunFecha(eServicio listaServicios[], int sizeServicios, eDiagnostico
listaDiagnosticos[], int sizeDiagnostico);
```

```
/// @brief Calcula y muestra la cotizacion promedio por mecanico
```

```
///
```

```
/// @param listaMecanicos array de lista de mecanicos hardcodeada
```

```
/// @param sizeMecanico largo de la lista de mecanicos (M)
```

```
/// @param listaServicios array de lista de servicios cargada por el usuario o hardcodeada
```

```
/// @param sizeServicios largo de la lista de servicios(T)
```

```
/// @param listaEspecialidades array de la lista de especialidades hardcodeada
```

```
/// @param sizeEspecialidad largo de la lista de especialidades (E)
```

```
void MostrarCotizacionPromedioPorMecanico(eMecanico listaMecanicos[], int sizeMecanico,
eServicio listaServicios[], int sizeServicio, sEspecialidad listaEspecialidades[], int
sizeEspecialidad);
```

```
/// @brief Muestra el mecanico con mas servicios diagnosticados
```

```
///
```

```
/// @param listaMecanicos array de lista de mecanicos hardcodeada
```

```
/// @param sizeMecanico largo de la lista de mecanicos (M)
```

```
/// @param listaEspecialidades array de la lista de especialidades hardcodeada
```

```
/// @param sizeEspecialidad largo de la lista de especialidades (E)
```

```
void MostrarMecanicosMasServicios(eMecanico listaMecanicos[], int sizeMecanico,
sEspecialidad listaEspecialidades[], int sizeEspecialidad);
```

```
/// @brief Muestra una lista de servicios entre Marzo y Mayo, segun determinada especialidad
```

```
///
```

```
/// @param listaServicios array de lista de servicios cargada por el usuario o hardcodeada
```

```
/// @param sizeServicios largo de la lista de servicios(T)
```

```
/// @param listaEspecialidades array de la lista de especialidades hardcodeada
```

```
/// @param sizeEspecialidad largo de la lista de especialidades (E)
```

```
/// @param listaMecanicos array de lista de mecanicos hardcodeada
```

```
/// @param sizeMecanico largo de la lista de mecanicos (M)
```

```
void ListarServiciosMarzoMayoPorEspecialidad(eServicio listaServicios[], int sizeServicio,
sEspecialidad listaEspecialidad[], int sizeEspecialidad, eMecanico listaMecanicos[], int
sizeMecanico);
```

```

/// @brief Calcula y muestra el porcentaje de servicios diagnosticados por mecanico, en
funcion
/// al total de servicios diagnosticados
///
/// @param listaServicios array de lista de servicios cargada por el usuario o hardcodeada
/// @param sizeServicios largo de la lista de servicios(T)
/// @param listaMecanicos array de lista de mecanicos hardcodeada
/// @param sizeMecanico largo de la lista de mecanicos (M)
/// @param listaEspecialidades array de la lista de especialidades hardcodeada
/// @param sizeEspecialidad largo de la lista de especialidades (E)
void ListarPromedioServiciosDiagnosticados(eServicio listaServicios[], int sizeServicio,
eMecanico listaMecanicos[], int sizeMecanico, sEspecialidad listaEspecialidad[], int
sizeEspecialidad);

```

```

/// @brief Enlista el top 3 de los desperfectos mas diagnosticados
///
/// @param listaServicios array de lista de servicios cargada por el usuario o hardcodeada
/// @param sizeServicios largo de la lista de servicios(T)
/// @param listaDiagnosticos array de la lista de diagnosticos harcodeada
/// @param sizeDiagnostico largo de la lista de diagnosticos (D)
/// @param listaAuxiliar lista que auxilia a la lista de diagnosticos
void ListarTopDesperfectos(eServicio listaServicios[], int sizeServicios, eDiagnostico
listaDiagnosticos[], int sizeDiagnostico, eAuxiliar listaAuxiliar[]);

```

```

/// @brief Muestra una lista de los servicios ordenados por la especialidad del mecanico
///
/// @param listaServicios array de lista de servicios cargada por el usuario o hardcodeada
/// @param sizeServicios largo de la lista de servicios(T)
/// @param listaMecanicos array de lista de mecanicos hardcodeada
/// @param sizeMecanico largo de la lista de mecanicos (M)
/// @param listaEspecialidades array de la lista de especialidades hardcodeada
/// @param sizeEspecialidad largo de la lista de especialidades (E)
void MostrarServiciosOrdenadosPorEspecialidad(eServicio listaServicios[], int sizeServicio,
eMecanico listaMecanicos[], int sizeMecanico, sEspecialidad listaEspecialidades[], int
sizeEspecialidad);

```

- **Biblioteca "mecanico.h":**

```

/// @brief Le asigna el valor de la lista de diagnosticos a la lista auxiliar
///
/// @param listaDiagnosticos array de la lista de diagnosticos harcodeada
/// @param sizeDiagnostico largo de la lista de diagnosticos (D)
/// @param listaAuxiliar lista que auxilia a la lista de diagnosticos
void InicializarListaAuxiliar(eDiagnostico listaDiagnosticos[], int sizeDiagnostico, eAuxiliar
listaAuxiliar[]);

```

```

/// @brief Muestra la lista del top 3 de los desperfectos mas diagnosticados
///
/// @param listaDiagnosticos array de la lista de diagnosticos harcodeada
/// @param sizeDiagnostico largo de la lista de diagnosticos (D)
/// @param listaAuxiliar lista que auxilia a la lista de diagnosticos

```

```
/// @param sizeAuxiliar largo de la lista auxiliar
void MostrarListaDesperfectos(eDiagnostico listaDiagnosticos[], int sizeDiagnostico, eAuxiliar
listaAuxiliar[], int sizeAuxiliar);
```

```
/// @brief Pide al usuario un id de mecanico
///
/// @param listaMecanicos array de lista de mecanicos hardcodeada
/// @param sizeMecanico largo de la lista de mecanicos (M)
/// @param listaEspecialidades array de la lista de especialidades hardcodeada
/// @param sizeEspecialidad largo de la lista de especialidades (E)
/// @return el id pedido
int PedirIdMecanico(eMecanico listaMecanicos[], int sizeMecanico, sEspecialidad
listaEspecialidades[], int sizeEspecialidad);
```

```
/// @brief Comprueba que el id este en la lista de mecanicos
///
/// @param listaMecanicos array de lista de mecanicos hardcodeada
/// @param sizeMecanico largo de la lista de mecanicos (M)
/// @param idRecibido id a comprobar
/// @return (-1) si no lo encuentro y si lo encuentro, la posicion del array(index)
int BuscarIdMecanico(eMecanico listaMecanicos[], int sizeMecanicos, int idRecibido);
```

```
/// @brief Muestra los datos ed un mecanico
///
/// @param unMecanico mecanico a mostrar
/// @param unaEspecialidad especialidad del mecanico
void MostrarUnMecanico (eMecanico unMecanico, sEspecialidad unaEspecialidad);
```

```
/// @brief Muestra todos los mecanicos de la lista de mecanicos
///
/// @param listaMecanicos array de lista de mecanicos hardcodeada
/// @param sizeMecanico largo de la lista de mecanicos (M)
/// @param listaEspecialidades array de la lista de especialidades hardcodeada
/// @param sizeEspecialidad largo de la lista de especialidades (E)
void MostrarListaMecanico (eMecanico listaMecanicos[], int sizeMecanico, sEspecialidad
listaEspecialidades[], int sizeEspecialidad);
```

```
/// @brief Solicita al usuario un id de diagnostico
///
/// @param listaDiagnosticos array de la lista de diagnosticos harcodeada
/// @param sizeDiagnostico largo de la lista de diagnosticos (D)
/// @return el id pedido
int PedirIdDiagnostico(eDiagnostico listaDiagnosticos[], int sizeDiagnostico);
```

```
/// @brief comprueba que el id este en la lista de diagnosticos
///
/// @param listaDiagnosticos array de la lista de diagnosticos harcodeada
/// @param sizeDiagnostico largo de la lista de diagnosticos (D)
/// @param id id a comprobar
/// @return (-1) si no lo encuentro y (1) si lo encuentro
int BuscarIdDiagnostico(eDiagnostico listaDiagnosticos[], int sizeDiagnostico, int id);
```

```

/// @brief Muestra los datos de un diagnostico
///
/// @param unDiagnostico diagnostico a mostrar
void MostrarUnDiagnostico(eDiagnostico unDiagnostico);

```

```

/// @brief Muestra todos los diagnosticos de la lista de diagnosticos
///
/// @param listaDiagnosticos array de la lista de diagnosticos harcodeada
/// @param sizeDiagnostico largo de la lista de diagnosticos (D)
void MostrarListaDiagnosticos(eDiagnostico listaDiagnosticos[], int sizeDiagnostico);

```

```

/// @brief Muestra una lista de los mecanicos sin servicios asignados
///
/// @param listaMecanicos array de lista de mecanicos hardcodeada
/// @param sizeMecanico largo de la lista de mecanicos (M)
/// @param listaEspecialidades array de la lista de especialidades hardcodeada
/// @param sizeEspecialidad largo de la lista de especialidades (E)
void MostrarMecanicosLibres(eMecanico listaMecanicos[], int sizeMecanico, sEspecialidad
listaEspecialidades[], int sizeEspecialidad);

```

```

/// @brief Ordena a los mecanicos alfabeticamente por especialidad
///
/// @param listaMecanicos array de lista de mecanicos hardcodeada
/// @param sizeMecanico largo de la lista de mecanicos (M)
/// @param listaEspecialidades array de la lista de especialidades hardcodeada
/// @param sizeEspecialidad largo de la lista de especialidades (E)
void OrdenarMecanicosPorEspecialidad(eMecanico listaMecanicos[], int sizeMecanico,
sEspecialidad listaEspecialidades[], int sizeEspecialidad);

```

```

/// @brief Muestra el top 3 de las especialidades mas estudiadas por los mecanicos
///
/// @param listaEspecialidades array de la lista de especialidades hardcodeada
/// @param sizeEspecialidad largo de la lista de especialidades (E)
/// @param listaAuxiliar lista que auxilia a especialidades
void MostrarListaEspecialidadesMasEstudiadas(sEspecialidad listaEspecialidades[], int
sizeEspecialidad, eAuxiliar listaAuxiliar[]);

```

```

/// @brief Enlista el top 3 de las especialidades mas estudiadas por los mecanicos
///
/// @param listaMecanicos array de lista de mecanicos hardcodeada
/// @param sizeMecanico largo de la lista de mecanicos (M)
/// @param listaEspecialidades array de la lista de especialidades hardcodeada
/// @param sizeEspecialidad largo de la lista de especialidades (E)
/// @param listaAuxiliar lista que auxilia a especialidades
void ListarTopEspecialidadesEstudiadas(eMecanico listaMecanicos[], int sizeMecanico,
sEspecialidad listaEspecialidades[], int sizeEspecialidad, eAuxiliar listaAuxiliar[]);

```

• **Biblioteca "especialidad.h":**

```

/// @brief muestra los datos de una especialidad

```



```

///
/// @param unaEspecialidad especialidad a mostrar
void MostrarUnaEspecialidad(sEspecialidad unaEspecialidad);

/// @brief Muestra las especialidades de la lista de especialidades
///
/// @param listaEspecialidades array de la lista de especialidades hardcodeada
/// @param sizeEspecialidad largo de la lista de especialidades (E)
void MostrarListaEspecialidades(sEspecialidad listaEspecialidades[], int sizeEspecialidad);

/// @brief Solicita al usuario un id de especialidades
///
/// @param listaEspecialidades array de la lista de especialidades hardcodeada
/// @param sizeEspecialidad largo de la lista de especialidades (E)
/// @return el id pedido
int PedirIdEspecialidad(sEspecialidad listaEspecialidades[], int sizeEspecialidad);

/// @brief comprueba que el id este en la lista de especialidades
///
/// @param listaEspecialidades array de la lista de especialidades hardcodeada
/// @param sizeEspecialidad largo de la lista de especialidades (E)
/// @param id id a comprobar
/// @return (-1) si no lo encuentro y si lo encuentra, la posicion del array(index)
int BuscarIdEspecialidad(sEspecialidad listaEspecialidades[], int sizeEspecialidad, int id);

/// @brief Ordena la especialidades alfabeticamente
///
/// @param listaEspecialidades array de la lista de especialidades hardcodeada
/// @param sizeEspecialidad largo de la lista de especialidades (E)
void OrdenarEspecialidadesAlfabeticamente(sEspecialidad listaEspecialidades[], int
sizeEspecialidad);

```

- **Biblioteca "taller.h":**

```

/// @brief Muestra los datos de un taller
///
/// @param unTaller taller a mostrar
void MostrarUnTaller(Taller unTaller);

/// @brief Muestra los talleres de la lista de talleres
///
/// @param listaTaller array de la lista de talleres hardcodeada
/// @param sizeTaller largo de la lista de talleres (L)
void MostrarListaDeTalleres(Taller listaTalleres[], int sizeTalleres);

/// @brief Ordena los talleres alfabeticamente
///
/// @param listaTaller array de la lista de talleres hardcodeada
/// @param sizeTaller largo de la lista de talleres (L)
void OrdenarTalleresAlfabeticamente(Taller listaTaller[], int sizeTaller);

```

```

/// @brief Ordena los mecanicos alfabeticamente segun las especialidades
///
/// @param listaMecanicos array de lista de mecanicos hardcodeada
/// @param sizeMecanico largo de la lista de mecanicos (M)
/// @param listaTaller array de la lista de talleres hardcodeada
/// @param sizeTaller largo de la lista de talleres (L)
void OrdenarMecanicosPorTaller(eMecanico listaMecanicos[], int sizeMecanico, Taller
listaTaller[], int sizeTaller);

/**
 * @brief Ordena la lista de servicios alfabeticamente por direccion de taller del mecanico
 *
 * @param listaServicios array de lista de servicios cargada por el usuario o hardcodeada
 * @param sizeServicio largo de la lista de servicios(T)
 * @param listaMecanicos array de lista de mecanicos hardcodeada
 * @param sizeMecanico largo de la lista de mecanicos (M)
 * @param listaTaller array de la lista de talleres hardcodeada
 * @param sizeTaller largo de la lista de talleres (L)
 * @param listaEspecialidad array de la lista de especialidades hardcodeada
 * @param sizeEspecialidad largo de la lista de especialidades (E)
 */
void OrdenarServiciosPorDireccionTaller(eServicio listaServicios[], int sizeServicio, eMecanico
listaMecanicos[], int sizeMecanico, Taller listaTaller[], int sizeTaller, sEspecialidad
listaEspecialidad[], int sizeEspecialidad);

```

Enlace a drive del video explicativo.

<https://drive.google.com/file/d/1-qmt-W-qpEbWR3hua2Z561sR1fYATu29/view?usp=sharing>

PD: Tarde unos minutitos de mas porque no se me subia el archivo a drive. En caso de que sea corregido, muchas gracias!