

Recuperatorio primer parcial

Antonio Aguirre

Nro legajo: 111239

1°C

Estructura agregada

```
typedef struct
{
    int isEmpty;
    int id; //PK
    char nombre[51];
    int turno;
}eVeterinario;
```

Esta estructura guarda los datos de los veterinarios que atienden el consultorio, el turno será un int que almacena 1 si es turno mañana y 0 si es turno tarde.

Al agregarse esta estructura se debió modificar:

```
typedef struct
{
    int isEmpty;
    int id; //PK
    int idMascota; //FK
    int idServicio; //FK
    eFecha fecha;
    int idVeterinario; //FK
}eTrabajo;
```

Y se modificaron las funciones:

```
eTrabajo IngresoTrabajo(eServicio[],int,int,eMascota[],int,eVeterinario[],int);
int AltaTrabajos(eTrabajo[],int,eServicio[],int,int,eMascota[],int,eVeterinario[],int);

int ModificacionTrabajo(eTrabajo[],int,eServicio[],int,eMascota[],int,eVeterinario[],int);

int BajaTrabajo(eTrabajo[],int,eMascota[],int,eServicio[],int,eVeterinario[],int);

void MostrarListaTrabajos(eTrabajo[],int,eMascota[],int,eServicio[],int,eVeterinario[],int);
```

Diagrama entidad-relación

TRABAJOS
isEmpty
Id (Primary key)
idMascota (Foreign Key)
idServicio (Foreign Key)
FECHA
idVeterinario (Foreign Key)

SERVICIOS
isEmpty
Id (Primary key)
descripción
precio

MASCOTAS
isEmpty
Id (Primary key)
nombreMascota
edad
raza

FECHA
dia
mes
anio

VETERINARIOS
isEmpty
Id (Primary key)
nombre
turno

Nuevos informes

Listado de trabajos realizados en el turno mañana

Muestra una lista con todos los trabajos realizados únicamente por veterinarios que trabajen en el turno mañana.

Listado de veterinarios con las mascotas que atendieron

Muestra una lista con todos los veterinarios y los nombres de las mascotas que atendieron.

Prototipo y documentación de funciones

int `MostrarMenu()`;

Muestra el menú principal y pide una opción al usuario.

Retorna Opcion.

int `BuscarLibre(eTrabajo[],int)`;

Busca el index de algún espacio disponible en la lista de trabajos.

Retorna indexDisponible.

eTrabajo `IngresoTrabajo(eServicio[],int,int,eMascota[],int,eVeterinario[],int)`;

Pide los datos de la nueva estructura al usuario.

Retorna eTrabajo(con los datos ingresados).

int `AltaTrabajos(eTrabajo[],int,eServicio[],int,int,eMascota[],int,eVeterinario[],int)`;

Verifica que haya lugar para ingresar registros y pide los datos o informa que no hay lugar

Retorna 0 si no hay espacio, 1 si se pudo agregar

int `ModificacionTrabajo(eTrabajo[],int,eServicio[],int,eMascota[],int,eVeterinario[],int)`;

Pide un id para luego modificar el campo que desee el usuario.

Retorna -

int `BajaTrabajo(eTrabajo[],int,eMascota[],int,eServicio[],int,eVeterinario[],int)`;

Pide y verifica un id para eliminar ese registro.

Retorna -

int `OrdenarTrabajoNombre(eTrabajo[],int,eMascota[],int)`;

Ordena la lista trabajos por nombre de las mascotas

Retorna -

int `OrdenarTrabajoAnio(eTrabajo[],int,eMascota[],int)`;

Ordena la lista trabajos por año y, de coincidir este, por nombre de mascota.

Retorna -

void `MostrarListaTrabajos(eTrabajo[],int,eMascota[],int,eServicio[],int,eVeterinario[],int)`;

Lista los trabajos realizados, obteniendo el nombre de mascota en vez de su ID, la descripción del servicio en vez de su ID y el nombre del veterinario en vez de su ID

int `MostrarPrecioTotal(eTrabajo[],int,eServicio[],int)`;

Acumula en una variable el costo de todos los trabajos realizados y lo imprime

Retorna -

```
int ServiciosConMasTrabajos(eTrabajo[],int,eServicio[],int);
```

Recorre las listas Trabajos y Servicios y obtiene en un contador la cantidad máxima de servicios realizados entre todos los trabajos, e imprime el/los servicios que se hayan realizado esa cantidad de veces.

Retorna -

```
int listarServiciosMascotas(eTrabajo[],int,eMascota[],int,eServicio[],int);
```

Lista los servicios y los nombres de las mascotas que se lo realizaron.

Retorna -

```
int PromedioEdadServicio(eTrabajo[],int,eMascota[],int,eServicio[],int);
```

Pide al usuario un servicio, e imprime el promedio de edades de las mascotas que se realizaron el mismo.

Retorna -

```
int TrabajosTurnoManana(eTrabajo[],int,eServicio[],int,eMascota[],int,eVeterinario[],int);
```

Lista los trabajos pero solo con los Veterinarios que trabajen en el turno mañana.

Retorna -

```
int
```

```
ListarVeterinariosMascotas(eTrabajo[],int,eServicio[],int,eMascota[],int,eVeterinario[],int);
```

Lista los veterinarios y los nombres de las mascotas que fueron atendidas por ellos.

Retorna -

Video explicativo

https://drive.google.com/file/d/1kJ4AQPQvOPJISdJxJAYU_71CT0fwNrYU/view?usp=sharing