



# Universidad Autónoma de Baja California

## Facultad de Ingeniería Arquitectura y Diseño

### Ingeniero en Software y tecnologías emergentes

**Materia:** Programación Estructurada / Clave 36276

**Alumno:** Antonio Ramos Gonzalez

**Matrícula:** 372576

**Maestro:** Pedro Núñez Yépiz

**Actividad No.** : 11 ANEXOS

**Tema - Unidad** : Librerías y Métodos de Ordenación

**Ensenada Baja California a 1 de noviembre del 2023**

```

// Antonio Ramos Gonzalez Mt 372576
// 01/11/2023
// Se generara un archivo donde se almacenara datos generados por el usuario
// RGA_Act11_932
#include "Curp.h"
#define N 2000
typedef struct _NombC
{
    char nombre[30];
    char nombre2[30];
    char apP[30];
    char apM[30];
} TNomb;
typedef struct _FrechaN
{
    int dia;
    int mes;
    int anio;
} TFecha;
typedef struct _Reg
{
    bool status = 1;
    int kay;
    TNomb nombre;
    TFecha fecha;
    char sex[2];
    char estado[30];
    char curp[18];
} Treg;

int msg();
int msg2();
void menu();
int verMt(Treg reg[], int n, int mt);
void estados(int est, char nombestado[]);
void _estadosAl(int est, char nombestado[]);
Treg agregarM(Treg reg[], int i);
Treg cargar(Treg reg[], int i);
void nombreAl(char nombre[], int sex);
void apAl(char apellido[]);
int month(int anio);
int day(int anio, int mes);
int searchSec(Treg reg[], int n, int mt);
int searchBin(Treg reg[], int inf, int sup, int mt);
int orderBu(Treg reg[], int n);
void pintReg(Treg reg[], int n);
void pintOneReg(Treg reg[], int n);
void swap(Treg reg[], int i, int j);
void createTXT(Treg reg[], int n);
int partition(Treg reg[], int Low, int high);
void quicksort(Treg reg[], int Low, int high);

int main()
{
    srand(time(NULL));
    fflush(stdin);
    menu();
    return 0;
}

int msg()
{
    system("CLS");
    printf("Menu\n");
    printf("1.-Agregar\n");
    printf("2.-Eliminar registro\n");
    printf("3.-Buscar Registro\n");
    printf("4.-Ordenar\n");
    printf("5.-Imprimir\n");
    printf("6.-Archivo texto\n");
    printf("0.-Salir\n");
}

```

```
printf("0.-Salir\n");|
return valid("Elije una opcion: ", 0, 6);
}
```

```
int msg2()
{
    system("CLS");
    printf("Menu\n");
    printf("1.-Agregar Manual\n");
    printf("2.-Cargar\n");
    printf("0.-Regresar\n");
    return valid("Elije una opcion: ", 0, 2);
}
```

```
void menu()
{
    int opc, elec, kay;
    int i = 0, j, band, bus;
    Treg reg[N];
    do
    {
        opc = msg();
        switch (opc)
        {
            case 1:
                elec = msg2();
                if (elec == 1)
                {
                    if (i < N)
                    {
                        reg[i] = agregarM(reg, i);
                        i++;
                        band = 1;
                    }
                    else
                    {
                        printf("Base de datos llena\n");
                        system("PAUSE");
                    }
                }
                if (elec == 2)
                {
                    if (i < N)
                    {
                        for (j = 0; j < 100; j++)
                        {
                            if (i < N)
                            {
                                reg[i] = cargar(reg, i);
                                i++;
                                band = 1;
                            }
                        }
                    }
                    else
                    {
                        printf("Base de datos llena\n");
                        system("PAUSE");
                    }
                }
            }

            break;
            case 2:
                kay = valid("Ingrese matricula que desea eliminar: ", 300000, 399999);
                if (band)
                {
                    bus = searchSec(reg, i, kay);
                    pintOneReg(reg, bus);
                }
                else
                {

```

```

    bus = searchBin(reg, 0, i, kay);
    pintOneReg(reg, bus);
}
if (valid("DESEA ELIMINAR ALUMNO\n1.-Si\n0.-NO\n", 0, 1))
{
    reg[bus].status = 0;
}

break;
case 3:
    kay = valid("Ingreso matricula que desea Buscar: ", 300000, 399999);
    if (band)
    {
        bus = searchSec(reg, i, kay);
        pintOneReg(reg, bus);
    }
    else
    {
        bus = searchBin(reg, 0, i, kay);
        pintOneReg(reg, bus);
    }
    system("PAUSE");
    break;
case 4:
    if (band)
    {
        if (i < 500)
        {
            band = orderBu(reg, i);
        }
        else
        {
            band = 1;
            quicksort(reg, 0, i);
        }
    }
    break;
case 5:
    pintReg(reg, i);
    break;
case 6:
    createTXT(reg, i);
    break;
}
} while (opc != 0);
}

```

Treg agregarM(Treg reg[], int i)

```

{
    char nomb[30], nomb2[30], apP[30], apM[30];
    char carp[18];
    int dia, mes, anio, bis, sex, est = 0;
    do
    {
        reg[i].kay = valid("Ingreso matricula: ", 300000, 399999);
    } while (verMt(reg, i, reg[i].kay));

    do
    {
        validCad("Ingreso nombre: ", nomb);
    } while (nomb[0] == '\0' || strlen(nomb) > 15);

    do
    {
        validCad("Ingreso 2do nombre(si tiene mas de 2 ingreselos aqui): ", nomb2);
    } while (strlen(nomb2) > 30);
    do
    {
        validCad("Ingreso apellido paterno: ", apP);
    } while (strlen(apP) > 15);
}

```

```

    } while (strlen(apP) > 15);
do
{
    validCad("Ingrese apellido materno: ", apM);
} while (strlen(apM) > 15);

system("CLS");
anio = valid("Ingresa anio de nacimiento: ", 1950, 2023);
if (anio == 2023)
{
    mes = valid("Ingresa mes de nacimiento(enero=01,diciembre=12): ", 1, 10);
}
else
{
    mes = valid("Ingresa mes de nacimiento(enero=01,diciembre=12): ", 1, 12);
}
if (mes == 1 || mes == 3 || mes == 5 || mes == 7 || mes == 8 || mes == 10 || mes == 12)
{
    dia = valid("Ingresa dia de nacimiento: ", 1, 31);
}
if (mes == 4 || mes == 6 || mes == 9 || mes == 11)
{
    dia = valid("Ingresa dia de nacimiento: ", 1, 30);
}
if (mes == 2)
{
    bis = bisiestro(anio);
    if (!bis)
    {
        dia = valid("Ingresa dia de nacimiento: ", 1, 28);
    }
    else
    {
        dia = valid("Ingresa dia de nacimiento: ", 1, 29);
    }
}

printf("1.-Hombre");
printf("2.-Mujer");
sex = valid("INGRESE SEXO: ", 1, 2);
imprimirEstados();
est = valid("Ingresa estado: ", 1, 33);
est--;
printf("%d", est);
system("PAUSE");
estados(est, reg[i].estado);
printf("%s", reg[i].estado);
system("PAUSE");
curp(carp, nomb, nomb2, apP, apM, dia, mes, anio, sex, est);
strcpy(reg[i].nombre.nombre, nomb);
strcpy(reg[i].nombre.nombre2, nomb2);
strcpy(reg[i].nombre.apP, apP);
strcpy(reg[i].nombre.apM, apM);
strcpy(reg[i].curp, carp);
reg[i].fecha.anio = anio;
reg[i].fecha.mes = mes;
reg[i].fecha.dia = dia;
if (sex == 1)
{
    strcpy(reg[i].sex, "H");
}
if (sex == 2)
{
    strcpy(reg[i].sex, "M");
}

return reg[i];
}

```

```

}

void estados(int est, char nombestado[])
{

```

```
void estados(int est, char nombestado[])
```

```
{
```

```
    char estados[33][30] = {
        "Aguascalientes",
        "Baja California",
        "Baja California Sur",
        "Campeche",
        "Chiapas",
        "Chihuahua",
        "Coahuila",
        "Colima",
        "Durango",
        "Guanajuato",
        "Guerrero",
        "Hidalgo",
        "Jalisco",
        "Estado de Mexico",
        "Michoacan",
        "Morelos",
        "Nayarit",
        "Nuevo Leon",
        "Oaxaca",
        "Puebla",
        "Queretaro",
        "Quintana Roo",
        "San Luis Potosi",
        "Sinaloa",
        "Sonora",
        "Tabasco",
        "Tamaulipas",
        "Tlaxcala",
        "Veracruz",
        "Yucatan",
        "Zacatecas",
        "Ciudad de Mexico",
        "Extranjero"};
    strcpy(nombestado, estados[est]);
```

```
}
```

```
void nombreAl(char nombre[], int sex)
```

```
{
```

```
    int i, al;
    char nombH[50][20] = {
        "JUAN", "PEDRO", "LUIS", "CARLOS", "ANDRES", "MIGUEL", "DAVID", "RAFAEL", "FRANCISCO",
        "ALEJANDRO", "JORGE", "DANIEL", "GUILLERMO", "SERGIO", "PABLO", "JAVIER", "ALBERTO",
        "SERGIO", "OSCAR", "GUILLERMO", "MARIO", "RICARDO", "IGNACIO", "FEDERICO", "RODRIGO",
        "MANUEL", "ANTONIO", "RAUL", "ALFREDO", "FERNANDO", "ROBERTO", "EDUARDO", "HUGO", "ENRIQUE",
        "EMILIO", "ARMANDO", "JULIO", "ADRIAN", "GUSTAVO", "ARTURO", "ANGEL", "HORACIO",
        "RODOLFO", "LUCAS", "RAMON", "OMAR", "CESAR", "SANTIAGO", "GERARDO", "ERNESTO"};
    char nombM[50][20] = {
        "ANA", "SOFIA", "CARMEN", "ISABEL", "ANDREA", "PATRICIA", "ELENA", "MARTA", "BEATRIZ",
        "PAULA", "JULIA", "VALERIA", "RAQUEL", "NATALIA", "CLAUDIA", "VALENTINA", "LORENA",
        "ELENA", "MARIANA", "ADRIANA", "LAURA", "ANTONIA", "MARTINA", "ANGELA", "LUISA",
        "SARA", "VERONICA", "ALICIA", "ROSA", "LUCIA", "VICTORIA", "GLORIA", "INES",
        "ROCIO", "VANESA", "RAFAELA", "ANAI", "EVA", "SANDRA", "LOLA", "ANA", "ESTHER",
        "MIRIAM", "LIDIA", "YOLANDA", "INES", "MARINA", "CAROLINA", "PILAR", "IRENE"};
    for (i = 0; i < 10; i++)
    {
        al = rand() % 50;
    }
    if (sex == 1)
    {
        strcpy(nombre, nombH[al]);
    }
    else
    {
        strcpy(nombre, nombM[al]);
    }
}
```

```
}
```

```

        strcpy(nombM, nombM[al]);
    }
}

void apAl(char apellido[])
{
    int i, al;
    char ap[87][20] =
    {
        "PEREZ", "LOPEZ", "GONZALEZ", "RODRIGUEZ", "MARTINEZ", "SANCHEZ", "FERNANDEZ", "GARCIA", "RODRIGUEZ", "DIAZ", "TORRES", "RAMIREZ", "JIMENEZ",
        "VARGAS", "MORALES", "RUIZ", "CASTRO", "HERRERA", "MEDINA", "NAVARRO", "DELGADO", "ROMERO", "MORENO", "GOMEZ", "VARGAS", "PEREZ", "FLORES", "GONZALEZ",
        "NUÑEZ", "SILVA", "SOTO", "MENDOZA", "ROJAS", "MENDEZ", "SALAS", "AGUIRRE", "BENITEZ", "GUZMAN", "PAREDES", "RIOS", "VALENZUELA", "CORDERO", "ESCOBAR",
        "OCHOA", "MORALES", "PIZARRO", "ARAYA", "CABRERA", "LOPEZ", "ORTEGA", "FERNANDEZ", "NAVARRO", "GUZMAN", "MENDEZ", "ROJAS", "SALAZAR", "VILLANUEVA", "PAREDES",
        "RIVAS", "SANDOVAL", "ESPINOZA", "MUNOZ", "ACOSTA", "DELGADO", "CASTRO", "NAVARRO", "GONZALEZ", "MENDOZA", "SILVA", "RIOS", "ROJAS", "AGUIRRE", "MORALES", "GOMEZ",
        "SOTO", "TORRES", "PEREZ", "MEDINA", "BENITEZ", "GARCIA", "VARGAS", "RODRIGUEZ", "RAMIREZ", "TORRES", "MORALES", "HERRERA", "JIMENEZ"};

    for (i = 0; i < 10; i++)
    {
        al = rand() % 87;
    }

    strcpy(apellido, ap[al]);
}

int month(int anio)
{
    if (anio == 2023)
    {
        return rand() % 11 + 1;
    }
    else
    {
        return rand() % 12 + 1;
    }
}

int day(int anio, int mes)
{
    int bis;

    if (mes == 1 || mes == 3 || mes == 5 || mes == 7 || mes == 8 || mes == 10 || mes == 12)
    {
        return rand() % 31 + 1;
    }

    if (mes == 4 || mes == 6 || mes == 9 || mes == 11)
    {
        return rand() % 30 + 1;
    }

    if (mes == 2)
    {
        bis = bisiesto(anio);
        if (!bis)
        {
            return rand() % 28 + 1;
        }
        else
        {
            return rand() % 29 + 1;
        }
    }
}

Treg cargar(Treg reg[], int i)
{
    char nomb[30], nomb2[30], apP[30], apM[30];
    int sex, est = 0;
    int n, j = 0;
    do
    {
        reg[i].kay = rand() % 100000 + 300000;
    } while (verMt(reg, i, reg[i].kay));

    sex = rand() % 2 + 1;
}

```

```

sex = rand() % 2 + 1;
if (sex == 1)
{
    strcpy(reg[i].sex, "H");
}
if (sex == 2)
{
    strcpy(reg[i].sex, "M");
}

nombreAl(nomb, sex);

strcpy(reg[i].nombre.nombre, nomb);
while (j < 10)
{
    n = rand() % 10 + 1;
    j++;
}

if (n % 2 == 0)
{
    nomb2[0] = '\0';
}
else
{
    nombreAl(nomb2, sex);
}
strcpy(reg[i].nombre.nombre2, nomb2);
apAl(apP);
strcpy(reg[i].nombre.apP, apP);
apAl(apM);
strcpy(reg[i].nombre.apM, apM);

reg[i].fecha.anio = 1950 + rand() % (2023 - 1950 + 1);
reg[i].fecha.mes = month(reg[i].fecha.anio);
reg[i].fecha.dia = day(reg[i].fecha.anio, reg[i].fecha.mes);

for (j = 0; j < 10; j++)
{
    est = rand() % 33;
}

estados(est, reg[i].estado);

curp(reg[i].curp, nomb, nomb2, apP, apM, reg[i].fecha.dia, reg[i].fecha.mes, reg[i].fecha.anio, sex, est);
return reg[i];
}

```

```

int searchSec(Treg reg[], int n, int mt)
{
    int i;
    i = 0;
    for (i = 0; i <= n; i++)
    {
        if (reg[i].kay == mt)
        {
            return i;
        }
    }
    return -1;
}

```

```

int searchBin(Treg reg[], int inf, int sup, int mt)
{
    int med; // variable local
    while (inf <= sup)
    {
        med = (inf + sup) / 2;
        if (reg[med].kay == mt)
        {
            return med;
        }
    }
}

```



```

    if (reg[med].kay == mt)
    {
        return med;
    }
    else
    {
        if (mt < reg[med].kay)
        {
            sup = med--;
        }
        else
        {
            inf = med++;
        }
    }
}

return -1;
}

int orderBu(Treg reg[], int n)
{
    int i, j;
    Treg temp;

    for (i = 0; i < n - 1; i++)
    {
        for (j = i + 1; j < n; j++)
        {
            if (reg[j].kay < reg[i].kay)
            {
                temp = reg[i];
                reg[i] = reg[j];
                reg[j] = temp;
            }
        }
    }
    return 0;
}

void pintReg(Treg reg[], int n)
{
    int i = 0, j = 0, elec;
    printf("%s", reg[n].sex);
    system("PAUSE");
    printf("%-6s %-15s %-23s %-15s %-10s %-10s %-10s %-10s \n",
           "No.", "Matricula", "Nombre", "ApP", "ApM", "Edad", "Sexo", "Curp");
    do
    {
        i = 0;
        if (j < n)
        {
            while (i < 40 && j < n)
            {
                if (reg[j].status)
                {
                    printf("%-6d %-10d %-10s %-15s %-15s %-13s %-10d %-5s \n",
                           j + 1, reg[j].kay, reg[j].nombre.nombre, reg[j].nombre.nombre2, reg[j].nombre.apP,
                           reg[j].nombre.apM, 2023 - reg[j].fecha.anio, reg[j].sex, reg[j].curp);
                }
                i++;
                j++;
            }
            elec = valid("1.-SI\n0.-NO\nSEGUIR IMPRIMIENDO: ", 0, 1);
        }
        else
        {
            printf("LIMITE ALCANZADO\n");
            elec = 0;
        }
    } while (elec);
    system("PAUSE");
}

```

```

system("PAUSE");
}

void pintOneReg(Treg reg[], int n)
{
    if (n != -1)
    {
        printf("MATRICULA: %d\nNOMBRE: %s %s\nAPELLIDO PATERNO: %s\nAPELLIDO MATERNO: %s\nFECHA DE NACIMIENTO: %02d-%02d-%d\nSEXO: %s\nESTADO: %s\nCURP: %s\n",
            reg[n].kay, reg[n].nombre.nombre, reg[n].nombre.nombre2, reg[n].nombre.apP,
            reg[n].nombre.apM, reg[n].fecha.mes, reg[n].fecha.dia, reg[n].fecha.anio, reg[n].sex, reg[n].estado, reg[n].curp);
    }
    else
    {
        printf("Registro no encontrado\n");
    }
}
}

```

```

int verMt(Treg reg[], int n, int mt)
{
    int i, j;
    for (i = 0; i < n; i++)
    {
        for (j = 0; j <= i; j++)
        {
            if (reg[j].kay == mt)
            {
                return 1;
            }
        }
    }

    return 0;
}

```

```

void createTXT(Treg reg[], int n)
{
    FILE *fa;
    int i = 0;
    fa = fopen("C:\\Users\\PC\\Desktop\\UABC 3er sem\\Programacion-estructurada\\Actividad_11\\Registros.txt", "w");
    fprintf(fa, "%-6s %-15s %-23s %-15s %-10s %-10s %-10s %s\n",
        "No.", "Matricula", "Nombre", "ApP", "ApM", "Edad", "Sexo", "Curp");
    for (i = 0; i < n; i++)
    {
        if (reg[i].status != 0)
        {
            fprintf(fa, "%-6d %-10d %-10s %-15s %-15s %-10s %02d-%02d-%8d %-5s %s\n",
                i + 1, reg[i].kay, reg[i].nombre.nombre, reg[i].nombre.nombre2, reg[i].nombre.apP,
                reg[i].nombre.apM, reg[i].fecha.mes, reg[i].fecha.dia, reg[i].fecha.anio, reg[i].sex, reg[i].curp);
        }
    }
    fclose(fa);
}

```

```

void swap(Treg reg[], int i, int j)
{
    Treg temp = reg[i];
    reg[i] = reg[j];
    reg[j] = temp;
}

```

```

int partition(Treg reg[], int Low, int high)
{
    Treg pivot;
    pivot.kay = reg[high].kay;
    int i = Low - 1;

    for (int j = Low; j <= high - 1; j++)
    {
        if (reg[j].kay <= pivot.kay)
        {
            i++;
            swap(reg, i, j);
        }
    }
    swap(reg, i + 1, high);
    return i + 1;
}

```

```
    {  
        i++;  
        swap(reg, i, j);  
    }  
}  
swap(reg, i + 1, high);  
return i + 1;  
}
```

```
void quicksort(Treg reg[], int Low, int high)  
{  
    if (Low < high)  
    {  
        int pi = partition(reg, Low, high);  
  
        quicksort(reg, Low, pi - 1);  
        quicksort(reg, pi + 1, high);  
    }  
}
```

## Menu

- 1.-Agregar
- 2.-Eliminar registro
- 3.-Buscar Registro
- 4.-Ordenar
- 5.-Imprimir
- 6.-Archivo texto
- 0.-Salir

Elije una opcion:

No.	Matricula	Nombre		ApP	ApM	Edad	Sexo	Curp
1	307706	OMAR		FERNANDEZ	AGUIRRE	8	H	FEA0151229HPLRGMB9
2	305848	INES	YOLANDA	RAMIREZ	GUZMAN	59	M	RAGI640625MSRMZN08
3	319253	SARA		MENDEZ	JIMENEZ	19	M	MEJS040807MASNMRA5
4	332162	PABLO	RAFAEL	RIOS	GARCIA	13	H	RIGP100913HSRSRBB9
5	314225	CLAUDIA	ANGELA	PAREDES	GONZALEZ	46	M	PAGC770717MSLRNL06
6	328101	MARIO		MORALES	GUZMAN	26	H	MOGM970422HSPRZR03
7	325167	ALEJANDRO		BENITEZ	PAREDES	9	H	BEPA140807HMCNRLB4
8	310927	SERGIO		ROMERO	ARAYA	65	H	ROAS580311HMSMYR03
9	331319	MANUEL	RODOLFO	NAVARRO	MARTINEZ	70	H	NAMM531222HTSVRN04
10	330683	ANTONIO		PIZARRO	ROJAS	53	H	PIRA701219HGTZJN07
11	314703	MARTA		ROJAS	BENITEZ	37	M	ROBM860303MNEJNR00
12	308336	OSCAR	LUIS	ACOSTA	MENDOZA	52	H	AOM0710320HQTCSN08
13	302850	HUGO	EMILIO	ROJAS	MORENO	32	H	ROMH911104HBSJRG09
14	328830	ARTURO		MORALES	MORALES	29	H	MOMA940303HCLRRR05
15	303441	RAUL	EMILIO	JIMENEZ	JIMENEZ	4	H	JIJR191229HSRMMLB2
16	327007	YOLANDA		MORENO	ROJAS	72	M	MORY511212MMCRCJL02
17	305111	EVA	ROSA	PAREDES	GOMEZ	41	M	PAGE820603MTLRMV09
18	327641	ROCIO		RIVAS	MORALES	60	M	RIMR630919MCLVRC03
19	325542	MARINA		RIOS	AGUIRRE	66	M	RIAM570809MHGSGR07
20	318856	MARIO		JIMENEZ	RUIZ	18	H	JIRM050816HHGMZRA1
21	302701	GLORIA	PILAR	MORALES	ORTEGA	72	M	MOOG510213MQRRRL00
22	305030	MIGUEL		MORALES	GARCIA	61	H	MOGM620625HPLRRG02
23	305766	VANESA	ELENA	GONZALEZ	FERNANDEZ	46	M	GOFV771021MTSNRN08
24	323626	VERONICA		DIAZ	CABRERA	10	M	DICV131025MSLZBRB9
25	308294	SERGIO	EMILIO	ROJAS	TORRES	18	H	ROTS051125HTCJRA7
26	318349	RODOLFO	OMAR	TORRES	MORALES	20	H	TOMR030513HPLRRDA0
27	303453	RAMON		GARCIA	MENDOZA	35	H	GAMR880503HNLRLNM09
28	303315	LOLA		ESCOBAR	GONZALEZ	48	M	EOGL750203MDGSNL07
29	332457	ELENA	ISABEL	SANCHEZ	CABRERA	35	M	SACE880217MMSNBL04
30	311515	IGNACIO		SANDOVAL	NUNEZ	41	H	SANI820614HQRNNG08
31	329562	SANTIAGO	ANGEL	SILVA	ROJAS	25	H	SIRS980201HMNLJN05
32	317993	PEDRO		TORRES	RIVAS	51	H	TORP721006HTLRVD07
33	316040	MARTINA	RAQUEL	OCHOA	JIMENEZ	53	M	00JM701025MGRCMR05

	No.	Matricula	Nombre		ApP	ApM	Edad	Sexo	Curp
2	1	307706	OMAR		FERNANDEZ	AGUIRRE	12-29-2015	H	FEA0151229HPLRGMB9
3	2	305848	INES	YOLANDA	RAMIREZ	GUZMAN	06-25-1964	M	RAGI640625MSRMZN08
4	3	319253	SARA		MENDEZ	JIMENEZ	08-07-2004	M	MEJS040807MASNMRA5
5	4	332162	PABLO	RAFAEL	RIOS	GARCIA	09-13-2010	H	RIGP100913HSRSRBB9
6	5	314225	CLAUDIA	ANGELA	PAREDES	GONZALEZ	07-17-1977	M	PAGC770717MSLRNL06
7	6	328101	MARIO		MORALES	GUZMAN	04-22-1997	H	MOGM970422HSPRZR03
8	7	325167	ALEJANDRO		BENITEZ	PAREDES	08-07-2014	H	BEPA140807HMCNRLB4
9	8	310927	SERGIO		ROMERO	ARAYA	03-11-1958	H	ROAS580311HMSMYR03
10	9	331319	MANUEL	RODOLFO	NAVARRO	MARTINEZ	12-22-1953	H	NAMM531222HTSVRN04
11	10	330683	ANTONIO		PIZARRO	ROJAS	12-19-1970	H	PIRA701219HGTZJN07
12	11	314703	MARTA		ROJAS	BENITEZ	03-03-1986	M	ROBM860303MNEJNR00
13	12	308336	OSCAR	LUIS	ACOSTA	MENDOZA	03-20-1971	H	AQMO710320HQTCSN08
14	13	302850	HUGO	EMILIO	ROJAS	MORENO	11-04-1991	H	ROMH911104HBSJRG09
15	14	328830	ARTURO		MORALES	MORALES	03-03-1994	H	MOMA940303HCLRRR05
16	15	303441	RAUL	EMILIO	JIMENEZ	JIMENEZ	12-29-2019	H	JIJR191229HSRMLB2
17	16	327007	YOLANDA		MORENO	ROJAS	12-12-1951	M	MORY511212MMCRJL02
18	17	305111	EVA	ROSA	PAREDES	GOMEZ	06-03-1982	M	PAGE820603MTLRMV09
19	18	327641	ROCIO		RIVAS	MORALES	09-19-1963	M	RIMR630919MCLVRC03
20	19	325542	MARTINA		RIOS	AGUIRRE	08-09-1957	M	RIAM570809MHGSGR07
21	20	318856	MARIO		JIMENEZ	RUIZ	08-16-2005	H	JIRM050816HHGMZRA1
22	21	302701	GLORIA	PILAR	MORALES	ORTEGA	02-13-1951	M	MOOG510213MQRRRL00
23	22	305030	MIGUEL		MORALES	GARCIA	06-25-1962	H	MOGM620625HPLRRG02
24	23	305766	VANESA	ELENA	GONZALEZ	FERNANDEZ	10-21-1977	M	GOFV771021MTSNRN08
25	24	323626	VERONICA		DIAZ	CABRERA	10-25-2013	M	DICV131025MSLZBRB9
26	25	308294	SERGIO	EMILIO	ROJAS	TORRES	11-25-2005	H	ROTS051125HTCJRR07
27	26	318349	RODOLFO	OMAR	TORRES	MORALES	05-13-2003	H	TOMR030513HPLRRDA0
28	27	303453	RAMON		GARCIA	MENDOZA	05-03-1988	H	GAMR880503HNLRNM09
29	28	303315	LOLA		ESCOBAR	GONZALEZ	02-03-1975	M	EOGL750203MDGSNL07
30	29	332457	ELENA	ISABEL	SANCHEZ	CABRERA	02-17-1988	M	SACE880217MMSNBL04
31	30	311515	IGNACIO		SANDOVAL	NUNEZ	06-14-1982	H	SANI820614HQRNNG08
32	31	329562	SANTIAGO	ANGEL	SILVA	ROJAS	02-01-1998	H	SIRS980201HMNLJN05
33	32	317993	PEDRO		TORRES	RIVAS	10-06-1972	H	TORP721006HTLRVD07
34	33	316040	MARTINA	RAQUEL	OCHOA	JIMENEZ	10-25-1970	M	OOJM701025MGRCMR05
35	34	312848	FERNANDO	RICARDO	GUZMAN	NAVARRO	10-01-2008	H	GUNF081001HMNZVRA8
36	35	310404	ROBERTO		NUNEZ	DELGADO	02-04-1967	H	MUDR670204HNTNLB02
37	36	322339	JORGE		MEDINA	RODRIGUEZ	03-12-1990	H	MERJ900312HVZDDR07
38	37	304090	HORACIO		RODRIGUEZ	MORALES	07-02-1963	H	ROMH630702HTLDRR04
39	38	319549	IGNACIO	MIGUEL	MUNOZ	CORDERO	10-28-1962	H	MUCI621028HASNRG06
40	39	313871	GUILLERMO		LOPEZ	FERNANDEZ	06-19-1973	H	LOFG730619HTSPRL08
41	40	301826	RAFAEL		MARTINEZ	MENDEZ	01-11-1993	H	MAMR930111HOCNRF03
42	41	325281	OMAR	IGNACIO	HERRERA	GUZMAN	04-28-1953	H	HEG0530428HMSRZM09
43	42	304720	LIDIA		NAVARRO	GONZALEZ	12-28-1989	M	NAGL891228MSRVND08
44	43	317190	SANTIAGO		RIOS	OCHOA	07-09-1953	H	RIOS530709HMSSCN04
45	44	322608	RAFAEL		MORALES	MORALES	06-21-1995	H	MOMR950621HPLRRF09
46	45	321738	CARMEN		CASTRO	DELGADO	02-23-1995	M	CADC950223MQRSLR04
47	46	321478	MARIO		VARGAS	MORALES	01-08-1962	H	VAMM620108HCMRRR02
48	47	316484	FERNANDO	GUILLERMO	GARCIA	GUZMAN	02-17-2008	H	GAGF080217HNLRZRA7
49	48	301134	PAULA	LORENA	TORRES	RAMIREZ	06-29-1988	M	TORP880629MMNRML01
50	49	320404	GERARDO		ESCOBAR	NAVARRO	10-18-1977	H	EONG771018HDGSVR04
51	50	316971	PAULA	ELENA	MORALES	JIMENEZ	07-23-2005	M	MOJP050723MQTRMLA7
52	51	314877	ARTURO		HERRERA	RODRIGUEZ	07-17-1995	H	HERA950717HGRRRDR06
53	52	302906	PAULA	SARA	GONZALEZ	SANCHEZ	02-15-1987	M	GOSP870215MZSNNL03
54	53	313081	MARIANA		DELGADO	SOTO	11-27-1982	M	DESM821127MDGLTR06
55	54	325068	YOLANDA		NUNEZ	HERRERA	07-20-1973	M	NUHY730720MCHNRL08
56	55	323685	ISABEL		SALAZAR	MORALES	07-28-2002	M	SAMI020728MPLLSA8
57	56	326443	ANGELA	CLAUDIA	SALAZAR	GOMEZ	01-02-2009	M	SAGA090102MNTLMNA4
58	57	326554	ANDRES	PABLO	ESCOBAR	MENDEZ	03-09-1990	H	EOMA900309HDGSNN07
59	58	329731	OSCAR	CARLOS	MUNOZ	SOTO	07-01-2020	H	MJSO200701HCCNTSC6