

Benemérita Universidad Autónoma de Puebla
Facultad de Ciencias de la Computación

PROGRAMACIÓN I

UNIDAD 2: ESTRUCTURAS



Docente:

Prof.^a Erika Bonfil Barragán



- EQUIPO 8

Jesús Huerta Aguilar		202041509
Javier De La Luz Ruiz		202033810
Ernesto Flores Cesáreo		202066335

Fecha de elaboración:

08/11/2021

NRC: 18438

Sección: 007

SEGUNDO SEMESTRE

Puebla, Pue.

Fecha de entrega: 29/11/2021

BENEMÉRITA UNIVERSIDAD AUTÓNOMA DE PUEBLA
FACULTAD DE CIENCIAS DE LA COMPUTACIÓN

1. Escriba un programa que lea dos fracciones e imprima la suma, resta, multiplicación y división de ambas fracciones.

Notas:

- a. Utilice una estructura para representar el dato fracción.
- b. Por cada operación implemente una función que reciba las fracciones con las que se operará y devuelva la fracción resultante.
- c. Imprima las fracciones resultantes de forma simplificada.

CODIGO:

```
//Jesús Huerta Aguilar, Javier de La Luz Ruiz, Ernesto Flores Cesareo.
/* 1. Escriba un programa que lea dos fracciones e imprima la suma, resta,
multiplicación y división de ambas fracciones. */
#include <stdio.h>
#include <stdlib.h>
#include <conio.h>

typedef struct
{
    int n;
    int d;
}fraccion;

void lectura(fraccion*,fraccion*);
fraccion suma(fraccion,fraccion,fraccion);
fraccion resta(fraccion,fraccion,fraccion);
fraccion producto(fraccion,fraccion,fraccion);
fraccion division(fraccion,fraccion,fraccion);
fraccion duroweon(fraccion);

int main(){
    fraccion f1, f2, f;
    lectura(&f1,&f2);
    f = suma(f1,f2,f);
    printf("La suma de fracciones es: %d/%d\n", f.n, f.d);
    f = resta(f1,f2,f);
    printf("La resta de fracciones es: %d/%d\n", f.n, f.d);
    f = producto(f1,f2,f);
    printf("El producto de fracciones es: %d/%d\n", f.n, f.d);
    f = division(f1,f2,f);
    printf("La division de fracciones es: %d/%d\n", f.n, f.d);
    getch();
    return 0;
}
```

```
}  
void lectura(fraccion *f1, fraccion *f2){  
    printf("Fraccion A");  
    printf("\nIngrese el valor del numerador en A: ");  
    scanf("%d", &f1->n);  
    printf("Ingrese el valor del denominador en A: ");  
    scanf("%d", &f1->d);  
    printf("\nFraccion B");  
    printf("\nIngrese el valor del numerador en B: ");  
    scanf("%d", &f2->n);  
    printf("Ingrese el valor del denominador en B: ");  
    scanf("%d", &f2->d);  
    system("cls");  
}  
fraccion suma(fraccion f1,fraccion f2,fraccion f){  
    int menor, aux, i, multiplo;  
    f.n=(f1.n*f2.d)+(f1.d*f2.n);  
    f.d=(f1.d*f2.d);  
    duroweon(f);  
}  
fraccion resta(fraccion f1,fraccion f2,fraccion f){  
    int menor, aux, i, multiplo;  
    f.n=(f1.n*f2.d)-(f1.d*f2.n);  
    f.d=(f1.d*f2.d);  
    duroweon(f);  
}  
fraccion producto(fraccion f1,fraccion f2,fraccion f){  
    int menor, aux, i, multiplo;  
    f.n=(f1.n*f2.n);  
    f.d=(f1.d*f2.d);  
    duroweon(f);  
}  
fraccion division(fraccion f1, fraccion f2,fraccion f){  
    int menor, aux, i, multiplo;  
    f.n=(f1.n*f2.d);  
    f.d=(f1.d*f2.n);  
    duroweon(f);  
}  
fraccion duroweon(fraccion f){  
    int menor, aux, i, multiplo;  
    do  
    {  
        if (f.n < f.d)  
        {  
            menor=f.n;  
        }else
```

```
{
    menor=f.d;
}
i=2;
multiplo=0;
while (i <= menor && multiplo ==0)
{
    if (f.n%i==0 && f.d%i==0)
    {
        aux=i;
        multiplo=1;
    }
    i++;
}
if (multiplo==1)
{
    f.n=f.n/aux;
    f.d=f.d/aux;
}
}while (multiplo==1);
return f;
}
```

VERSION PUNTO:

```
//Jesús Huerta Aguilar, Javier de La Luz Ruiz, Ernesto Flores Cesareo.
/* 1. Escriba un programa que Lea dos fracciones e imprima la suma, resta,
multiplicación y división de ambas fracciones. */
#include <stdio.h>
#include <stdlib.h>
#include <conio.h>

typedef struct
{
    int n;
    int d;
}fraccion;

void lectura(fraccion*,fraccion*);
void suma(fraccion,fraccion,fraccion*);
void resta(fraccion,fraccion,fraccion*);
void producto(fraccion,fraccion,fraccion*);
void division(fraccion,fraccion,fraccion*);
void duroweon(fraccion*);

int main(){
    fraccion f1,f2,f;
    lectura(&f1,&f2);
    suma(f1,f2,&f);
    resta(f1,f2,&f);
    producto(f1,f2,&f);
    division(f1,f2,&f);
    getch();
    return 0;
}

void lectura(fraccion *f1, fraccion *f2){
    printf("Fraccion A");
    printf("\nIngrese el valor del numerador en A: ");
    scanf("%d", &f1->n);
    printf("Ingrese el valor del denominador en A: ");
    scanf("%d", &f1->d);
    printf("\nFraccion B");
    printf("\nIngrese el valor del numerador en B: ");
    scanf("%d", &f2->n);
    printf("Ingrese el valor del denominador en B: ");
    scanf("%d", &f2->d);
    system("cls");
}

void suma(fraccion f1, fraccion f2, fraccion *f){
    f->n=(f1.n*f2.d)+(f1.d*f2.n);
    f->d=(f1.d*f2.d);
```

```
    duroweon(f);
    printf("La suma de fracciones es: %d/%d\n", f->n, f->d);
}
void resta(fraccion f1, fraccion f2, fraccion *f){
    f->n=(f1.n*f2.d)-(f1.d*f2.n);
    f->d=(f1.d*f2.d);
    duroweon(f);
    printf("La resta de fracciones es: %d/%d\n", f->n, f->d);
}
void producto(fraccion f1, fraccion f2, fraccion *f){
    f->n=(f1.n*f2.n);
    f->d=(f1.d*f2.d);
    duroweon(f);
    printf("El producto de fracciones es: %d/%d\n", f->n, f->d);
}
void division(fraccion f1, fraccion f2, fraccion *f){
    f->n=(f1.n*f2.d);
    f->d=(f1.d*f2.n);
    duroweon(f);
    printf("El producto de fracciones es: %d/%d\n", f->n, f->d);
}
void duroweon(fraccion *f){
    int menor, aux, i, multiplo;
    do
    {
        if (f->n < f->d)
        {
            menor=f->n;
        }else
        {
            menor=f->d;
        }
        i=2;
        multiplo=0;
        while (i <= menor && multiplo ==0)
        {
            if (f->n%i==0 && f->d%i==0)
            {
                aux=i;
                multiplo=1;
            }
            i++;
        }
        if (multiplo==1)
        {
            f->n=f->n/aux;

```

```
        f->d=f->d/aux;  
    }  
}while (multiplo==1);  
}
```

2. Un punto en el plano se puede representar mediante una estructura con dos campos. Escriba un programa que lea dos puntos en el plano e imprima la distancia y el punto medio entre ellos. Para ello implemente las siguientes dos funciones:

- a. Calcular distancia: La función recibirá dos parámetros de tipo Punto y devolver la distancia como flotante.
- b. Obtener punto medio: La función recibirá dos parámetros de tipo Punto, y retornará el punto medio.

```
//Jesús Huerta Aguilar, Javier de La Luz Ruiz, Ernesto Flores Cesareo
//Programación I - "Programa: Distancia entre dos puntos y punto medio"
///////////////////////////////// VERSIÓN PUNTO ///////////////////////////////////
//CARGAR LIBRERIAS
#include <stdio.h>
#include <stdlib.h>
#include <conio.h>
#include <string.h>
#include <math.h>
//ESTRUCTURAS
typedef struct{
    float x;
    float y;
}valores;
//PROTOTIPOS
void super();
void lect(valores, valores, valores);
float dist(valores, valores);
valores medio(valores, valores, valores);
void cont(char *);
//PRINCIPAL
int main(){
    char op;
    valores a, b, m;
    do{
        lect(a, b, m);
        cont(&op);
    } while (op == 'S' || op == 's');
    getch();
    return 0;
}
//NOTAS
void super(){
```


VERSION FLECHA:

```
//Jesús Huerta Aguilar, Javier de La Luz Ruiz, Ernesto Flores Cesareo
//Programación I - "Programa: Distancia entre dos puntos y punto medio"
//////////////////// VERSIÓN FLECHA //////////////////////
//CARGAR LIBRERIAS
#include <stdio.h>
#include <stdlib.h>
#include <conio.h>
#include <string.h>
#include <math.h>
//ESTRUCTURAS
typedef struct{
    float x;
    float y;
}valores;
//PROTOTIPOS
void super();
void lect(valores *, valores *);
float dist(valores, valores);
valores medio(valores, valores, valores *);
void cont(char *);
//PRINCIPAL
int main(){
    char op;
    valores a,b,m;
    do{
        lect(&a,&b);
        printf("\n\n\tDISTANCIA ENTRE A Y B:\tD = %f\n",dist(a,b));
        m = medio(a,b,&m);
        printf("\n\tPUNTO MEDIO A Y B:\tC(%.2f,%.2f)\n\n",m.x,m.y);
        cont(&op);
    } while (op == 'S' || op == 's');
    getch();
    return 0;
}
//NOTAS
void super(){
    int i;
    printf("\t\t");
    for ( i = 0; i < 53; i++){
```

```
        printf("=");
    }
    printf("\n");
    printf("\t\t\t\t\t PROGRAMACI%cN I\t\t\t\t\t EQUIPO 8\n",224);
    printf("\n\t\t\t\t\t - - - DISTANCIA Y PUNTO MEDIO ENTRE DOS PUNTOS - - -\n");
    printf("\t\t\t\t\t VERSI%cN FLECHA\n",224);
    printf("\t\t\t");
    for ( i = 0; i < 53; i++){
        printf("=");
    }
    printf("\n");
}
//LECTURA
void lect(valores *a, valores *b){
    int i;
    char l;
    for (i = 0; i < 2; i++){
        super();
        printf("\n%c %c %c %c INGRESA LOS VALORES DE LOS
PUNTOS\n",219,219,219,219);
        switch (i){
            case 0:
                l = 'A';
                break;
            case 1:
                l = 'B';
                break;
        }
        printf("\n\t\t\t\t\t VALORES DEL PUNTO %c:\n\n",l);
        printf("\t\t\t\t\t X = ");
        if (i == 0){
            scanf("%f",&a->x);
            printf("\t\t\t\t\t Y = ");
            scanf("%f",&a->y);
        }
        else{
            scanf("%f",&b->x);
            printf("\t\t\t\t\t Y = ");
            scanf("%f",&b->y);
        }
        system("cls");
    }
    super();
    printf("\n\t\t\t\t\t %c%c A(%.2f,%.2f)\t\t\t\t\t B(%.2f,%.2f) %c%c\n",219,219,a->x,a->y,b->x,b->y,219,219);
}
```

```
//CALCULAR DISTANCIA
float dist(valores a,valores b){
    float res;
    res = sqrt(((b.x-a.x)*(b.x-a.x))+((b.y-a.y)*(b.y-a.y)));
    return res;
}

//CALCULAR PUNTO MEDIO
valores medio(valores a,valores b,valores *m){
    m->x = (a.x+b.x)/2;
    m->y = (a.y+b.y)/2;
    return *m;
}

//CONTINUAR
void cont(char *op){
    int error;
    do{
        error = 0;
        printf("\n%c %c %c %c %cDESEA CONTINUAR? [S/N] >> ",219,219,219,219,168);
        scanf("%s",op);
        if (*op == 's' || *op == 'S'){
            system("cls");
        }
        else{
            if (*op == 'n' || *op == 'N'){
                system("cls");
                printf("\n\n\tGRACIAS POR USAR NUESTROS SERVICIOS");
                printf("\n\n\t\tEquipo 8\t-\tBUAP");
            }
            else{
                printf("\n\t\t\t\t[!] ERROR: Ingresa una opci%cn valida [!]\n",162);
                error = 1;
            }
        }
    } while (error == 1);
}
```

3. Una línea recta se puede representar mediante una estructura con dos campos de tipo Punto. Escriba un programa en C que lea los dos puntos por lo que pasa una recta e imprima la pendiente de la recta y su ecuación en la forma pendiente - ordenada al origen. Para ello implemente las siguientes funciones:

- a. float pendienteRecta(Recta "p).
- b. void ecuacionRecta(Recta "p).

CODIGO:

```
//Jesús Huerta Aguilar, Javier de La Luz Ruiz, Ernesto Flores Cesareo.
/* 3. Escriba un programa en C que lea los dos puntos por lo que pasa
una recta e imprima la pendiente de la recta y su ecuación en la forma pendiente
- ordenada al origen. */
#include <stdio.h>
#include <stdlib.h>
#include <conio.h>

typedef struct
{
    int x;
    int y;
}punto;

typedef struct
{
    punto p1;
    punto p2;
}recta;

void lectura(recta,float);
float pendienteRecta(recta);
void ecuacionRecta(recta,float);

int main(){
    recta r;
    float m;
    lectura(r,m);
    getch();
    return 0;
}

void lectura(recta r, float m){
    printf("Ingrese los valores del punto 1 x ");
    scanf("%d", &r.p1.x);
    printf("Ingrese los valores del punto 1 y ");
```

```
scanf("%d", &r.p1.y);
printf("Ingrese los valores del punto 2 x ");
scanf("%d", &r.p2.x);
printf("Ingrese los valores del punto 2 y ");
scanf("%d", &r.p2.y);
printf("\nLa formula de la pendiente es: ");
printf("\nm=(y2)-(y1)/(x2)-(x1) ----> m=(%d)-(%d)/(%d)-(%d)",r.p2.y,r.p1.y,r.p2.x,r.p1.x);
m = pendienteirecta(r);
printf("\nLa pendiente es %.2f",m);
ecuacionrecta(r,m);
}

float pendienteirecta(recta r){
    float pendiente;
    return pendiente = (float)(r.p2.y - r.p1.y)/(r.p2.x - r.p1.x);
}

void ecuacionrecta(recta r,float m){
    float b;
    printf("\n\nLa formula de b de pendiente interseccion es: ");
    printf("\nb=(y1)-((m)(x1)) ----> b=(%d)-((%.2f)(%d))",r.p1.y,m,r.p1.x);
    b=(r.p1.y-(m*r.p1.x));
    printf("\nLa ecuacion de forma pendiente ordenada al origen es:");
    if (b >= 0)
    {
        printf(" y = %.2fx+%.2f",m,b);
    }else
    {
        printf(" y = %.2fx%.2f",m,b);
    }
}
```

VERSION FLECHA:

```
//Jesús Huerta Aguilar, Javier de La Luz Ruiz, Ernesto Flores Cesareo.
/* 3. Escriba un programa en C que lea los dos puntos por lo que pasa
una recta e imprima la pendiente de la recta y su ecuación en la forma pendiente
- ordenada al origen. */
#include <stdio.h>
#include <stdlib.h>
#include <conio.h>

typedef struct
{
    int x;
    int y;
}punto;

typedef struct
{
    punto p1;
    punto p2;
}recta;

void lectura(recta*);
float pendiente(recta*);
void ecuacionrecta(recta*,float);

int main(){
    recta r;
    float m;
    lectura(&r);
    m = pendiente(&r);
    printf("La pendiente es %.2f",m);
    ecuacionrecta(&r,m);
    getch();
    return 0;
}

void lectura(recta *r){
    printf("Ingrese los valores del punto 1 x ");
    scanf("%d", &r->p1.x);
    printf("Ingrese los valores del punto 1 y ");
    scanf("%d", &r->p1.y);
    printf("Ingrese los valores del punto 2 x ");
    scanf("%d", &r->p2.x);
    printf("Ingrese los valores del punto 2 y ");
    scanf("%d", &r->p2.y);
}
```



```
printf("\nLa formula de la pendiente es: ");
printf("\nm=(y2)-(y1)/(x2)-(x1) ----> m=(%d)-(%d)/(%d)-(%d)",r->p2.y,r->p1.y,r-
>p2.x,r->p1.x);
}
float pendienterecta(recta *r){
    float pendiente;
    return pendiente = (float)(r->p2.y - r->p1.y)/(r->p2.x - r->p1.x);
}
void ecuacionrecta(recta *r,float m){
    float b;
    printf("\n\nLa formula de b de pendiente interseccion es: ");
    printf("\nb=(y1)-((m)(x1)) ----> b=(%d)-((%.2f)(%d))",r->p1.y,m,r->p1.x);
    b=(r->p1.y-(m*r->p1.x));
    printf("\nLa ecuacion de forma pendiente ordenada al origen es:");
    if (b >= 0)
    {
        printf(" y = %.2fx+%.2f",m,b);
    }else
    {
        printf(" y = %.2fx%.2f",m,b);
    }
}
```

BENEMÉRITA UNIVERSIDAD AUTÓNOMA DE PUEBLA
FACULTAD DE CIENCIAS DE LA COMPUTACIÓN

4. Escriba un programa que implemente una agenda que permita realizar lo siguiente:

a. Dar de alta a una persona en la agenda.

Benemérita Universidad Autónoma de Puebla

Estructuras y arreglos de estructuras|2

b. Eliminar a una persona de la agenda.

c. Buscar una persona de la agenda con base en su número telefónico.

d. Listar a todas las personas de la agenda.

e. Salir.

Nota: De cada persona se deben guardarlos siguientes datos: nombre, apellidos, número telefónico, correo electrónico y fecha de nacimiento.

NOTA: De cada programa se deben realizar dos versiones, en la primera se debe acceder a las estructuras utilizando el operador (.) y en la segunda, se debe utilizar el operador puntero (->).

48.2	8	20.5
------	---	------

CODIGO:

```
//4. Agenda version normal

//Librerias
#include <stdio.h>
#include <conio.h>
#include <stdlib.h>
#include <string.h>
//Estructura de datos
typedef struct{
    int dia,mes,anyo;
}Fecha;
typedef struct{
    char nombre[15];
    char apellido[30];
    char telefono[15];
    char correo[40];
    Fecha fecha;
}agenda;
agenda agen[100];
int cont=0;
//Prototipos
void nota();
void alta();
void eliminar();
void buscar();
void lista();
void final(char *,int);
//Principal
int main (){
```



```
system("cls");
int error;
nota();
printf("\n%c Nombre de la persona: ",219);
fflush(stdin);
gets(agen[cont].nombre);
system("cls");
nota();
printf("\n%c Apellidos de la persona: ",219);
fflush(stdin);
gets(agen[cont].apellido);
system("cls");
nota();
do{
    error=0;
    printf("\n%c N%cmero telef%cnico: ",219,163,162);
    fflush(stdin);
    gets(agen[cont].telefono);
    if (strlen(agen[cont].telefono)==10){
        system("cls");
    }else{
        error=1;
        printf("[!] ERROR: N%cmero telef%cnico inv%clido [!]",163,162,160);
    }
}while(error!=0);
nota();
printf("\n%c Correo electr%cnico: ",219,162);
fflush(stdin);
gets(agen[cont].correo);
system("cls");
nota();
printf("\n%c Fecha de Nacimiento:",219);
do{
    error=0;
    printf("\n\n%c D%ca: ",175,161);
    scanf("%d",&agen[cont].fecha.dia);
    if (agen[cont].fecha.dia<1 || agen[cont].fecha.dia>31){
        printf("[!] ERROR: D%ca ingresado inv%clido [!]",161,160);
        error=1;
    }
}while(error!=0);
do{
    error=0;
    printf("\n\n%c Mes: ",175);
    scanf("%d",&agen[cont].fecha.mes);
    if (agen[cont].fecha.mes<1 || agen[cont].fecha.mes>12){
```

```
        printf("[!] ERROR: Mes ingresado inválido [!]",160);
        error=1;
    }
}while(error!=0);
do{
    error=0;
    printf("\n\n%c A%co: ",175,164);
    scanf("%d",&agen[cont].fecha.anyo);
    if (agen[cont].fecha.anyo<1941 || agen[cont].fecha.anyo>2013){
        printf("[!] ERROR: A%co ingresado inválido [!]",164,160);
        error=1;
    }
}while(error!=0);

    cont++;
}

void eliminar(){
    system("cls");
    int pos;
    int i;
    if (cont == 0){
        printf("\n\t[!] ERROR: No hay contactos registrados [!]\n");
    }
    else{
        printf("\n\tNOTA:Puedes consultar el n%cmero que necesitas en la opci%cn
%clicar...%c\n",163,162,34,34);
        printf("\nIngresa el n%cmero de contacto en la agenda para eliminar:
",163,162);
        scanf("%d",&pos);
        if (pos-1 < cont){
            if (pos-1 == cont-1){
                cont -=1;
            }
            else{
                for (i = pos-1; i < cont; i++){
                    strcpy(agen[i].nombre,agen[i+1].nombre);
                    strcpy(agen[i].apellido,agen[i+1].apellido);
                    strcpy(agen[i].telefono,agen[i+1].telefono);
                    strcpy(agen[i].correo,agen[i+1].correo);
                    agen[i].fecha.dia=agen[i+1].fecha.dia;
                    agen[i].fecha.mes=agen[i+1].fecha.mes;
                    agen[i].fecha.anyo=agen[i+1].fecha.anyo;
                }
                cont-=1;
            }
        }
        printf("\n%c Datos ingresados eliminados.\n",219);
    }
}
```

```
    }
    else
    {
        printf("\n\t[!] ERROR: No hay contactos registrados [!]\n");
    }
}

void buscar (){
    system ("cls");
    char num [10];
    int i,aux=0;
    printf("Ingrese el número telefónico de la persona a buscar: ",163,162);
    fflush(stdin);
    gets(num);
    for(i = 0; i < cont; i++){
        if (strcmp(num,agen[i].telefono)==0){
            printf("\n%c%c%c%c Persona registrada #d
%c%c%c%c",205,205,205,205,i+1,205,205,205,205);
            printf("\nNombre completo: %s %s\n",agen[i].nombre,agen[i].apellido);
            printf("Número Teléfónico: %s\n",163,162,agen[i].telefono);
            printf("Correo electrónico: %s\n",162,agen[i].correo);
            printf("Fecha de Nacimiento:
%02d/%02d/%04d\n",agen[i].fecha.dia,agen[i].fecha.mes,agen[i].fecha.anyo);
            aux=1;
        }
    }
    if(aux==0){
        printf("\n%c Datos ingresados no registrados.\n",219);
    }
}

void lista (){
    system ("cls");
    int i;
    printf("\n      %CAGENDA%C\n",207,207);
    if (cont == 0){
        printf("\n\t[!] ERROR: No hay contactos registrados [!]\n");
    }
    else{
        for(i=0;i<cont;i++){
            printf("\n%c%c%c%c Persona registrada #d
%c%c%c%c",205,205,205,205,i+1,205,205,205,205);
            printf("\nNombre completo: %s %s\n",agen[i].nombre,agen[i].apellido);
            printf("Número Teléfónico: %s\n",163,162,agen[i].telefono);
            printf("Correo electrónico: %s\n",162,agen[i].correo);
            printf("Fecha de Nacimiento:
%02d/%02d/%04d\n",agen[i].fecha.dia,agen[i].fecha.mes,agen[i].fecha.anyo);
```

```
    }  
  }  
}  
void final(char *op2,int op){  
  int error;  
  do{  
    error = 0;  
    if (op == 5){  
      *op2 = 'N';  
    }  
    else{  
      printf("\n%cDesea realizar otra operaci%c\n dentro de la agenda? (s/n) %c",168,162,175);  
      scanf("%s",op2);  
    }  
    if (*op2 == 's' || *op2 == 'S'){  
      system("cls");  
    }  
    else{  
      if (*op2 == 'n' || *op2 == 'N'){  
        system("cls");  
        printf("\n\n\tGRACIAS POR USAR NUESTROS SERVICIOS");  
        printf("\n\n\t  Equipo 8\t-\t\tBUAP");  
      }  
      else{  
        printf("\n[!] ERROR: Ingresa una opci%c\n valida [!]",162);  
        error = 1;  
      }  
    }  
  } while (error!=0);  
}
```



```
printf ("\n\t%c[3] Buscar a una persona.",175);
printf ("\n\t%c[4] Listar a todos en la agenda.",175);
printf ("\n\t%c[5] Salir.",175);
printf ("\n\n=");
scanf ("%d",&op);
if (op<1 || op>5){
    printf ("\n[!] ERROR: Opci%cn no v%clida [!]",162,160);
    error=1;
    getch();
    system("cls");
}
}while(error!=0);
switch(op){
    case 1:
        alta(agen);
        break;
    case 2:
        eliminar(agen);
        break;
    case 3:
        buscar(agen);
        break;
    case 4:
        lista(agen);
        break;
}
    final(&op2,op);
}while(op2 == 's' || op2 == 'S');
getch();
return 0;
}
void nota (){
    printf("Ingresando los datos de la persona n%cmerno %d\n",163,cont+1);
}
void alta (agenda *A){
    system("cls");
    int error;
    nota();
    printf("\n%c Nombre de la persona: ",219);
    fflush(stdin);
    scanf ("%s", (A+cont)->nombre);
    //gets(A[cont]->nombre);
    system("cls");
    nota();
    printf("\n%c Apellidos de la persona: ",219);
    fflush(stdin);
```

```
scanf("%s", (A+cont)->apellido);
//gets(A[cont]->apellido);
system("cls");
nota();
do{
    error=0;
    printf("\n%c N%cmero telef%cnico: ", 219, 163, 162);
    fflush(stdin);
    scanf("%s", (A+cont)->telefono);
    //gets(A[cont]->telefono);
    if (strlen((A+cont)->telefono)==10){
        system("cls");
    }else{
        error=0;
        printf("[!] ERROR: N%cmero telef%cnico inv%clido [!]", 163, 162, 160);
    }
}while(error!=0);
nota();
printf("\n%c Correo electr%cnico: ", 219, 162);
fflush(stdin);
scanf("%s", (A+cont)->correo);
//gets(A[cont]->correo);
system("cls");
nota();
printf("\n%c Fecha de Nacimiento:", 219);
do{
    error=0;
    printf("\n\n%c D%ca: ", 175, 161);
    scanf("%d", &(A+cont)->fecha.dia);
    if ((A+cont)->fecha.dia<1 || (A+cont)->fecha.dia>31){
        printf("[!] ERROR: D%ca ingresado inv%clido [!]", 161, 160);
        error=1;
    }
}while(error!=0);
do{
    error=0;
    printf("\n\n%c Mes: ", 175);
    scanf("%d", &(A+cont)->fecha.mes);
    if ((A+cont)->fecha.mes<1 || (A+cont)->fecha.mes>12){
        printf("[!] ERROR: Mes ingresado inv%clido [!]", 160);
        error=1;
    }
}while(error!=0);
do{
    error=0;
    printf("\n\n%c A%co: ", 175, 164);
```

```
scanf("%d",&(A+cont)->fecha.anyo);
if ((A+cont)->fecha.anyo<1941 || (A+cont)->fecha.anyo>2013){
    printf("[!] ERROR: A%co ingresado inv%clido [!]",164,160);
    error=1;
}
}while(error!=0);
cont++;
}
void eliminar(agenda *agen){
    system("cls");
    int pos;
    int i;
    if (cont == 0){
        printf("\n\t[!] ERROR: No hay contactos registrados [!]\n");
    }
    else{
        printf("\n\tNOTA:Puedes consultar el n%cmero que necesitas en la opci%cn
%clicar...%c\n",163,162,34,34);
        printf("\nIngresa el n%cmero de contacto en la agenda para eliminar:
",163,162);
        scanf("%d",&pos);
        if (pos-1 < cont){
            if (pos-1 == cont-1){
                cont -=1;
            }
            else{
                for (i = pos-1; i < cont; i++){
                    strcpy((agen+i)->nombre,(agen+i+1)->nombre);
                    strcpy((agen+i)->apellido,(agen+i+1)->apellido);
                    strcpy((agen+i)->telefono,(agen+i+1)->telefono);
                    strcpy((agen+i)->correo,(agen+i+1)->correo);
                    (agen+i)->fecha.dia=(agen+i+1)->fecha.dia;
                    (agen+i)->fecha.mes=(agen+i+1)->fecha.mes;
                    (agen+i)->fecha.anyo=(agen+i+1)->fecha.anyo;
                }
                cont-=1;
            }
            printf("\n%c Datos ingresados eliminados.\n",219);
        }
        else
        {
            printf("\n\t[!] ERROR: No hay contactos registrados [!]\n");
        }
    }
}
void buscar (agenda *agen){
```

```
system ("cls");
char num [10];
int i,aux=0;
printf("Ingrese el número telefónico de la persona a buscar: ",163,162);
fflush(stdin);
gets(num);
for(i = 0; i < cont; i++){
    if (strcmp(num,(agen+i)->telefono)==0){
        printf("\n%c%c%c%c Persona registrada #d
%c%c%c%c",205,205,205,205,i+1,205,205,205,205);
        printf("\nNombre completo: %s %s\n",(agen+i)->nombre,(agen+i)->apellido);
        printf("Número Teléfónico: %s\n",163,162,(agen+i)->telefono);
        printf("Correo electrónico: %s\n",162,(agen+i)->correo);
        printf("Fecha de Nacimiento: %02d/%02d/%04d\n",(agen+i)-
>fecha.dia,(agen+i)->fecha.mes,(agen+i)->fecha.anyo);
        aux=1;
    }
}
if(aux==0){
    printf("\n%c Datos ingresados no registrados.\n",219);
}
}

void lista (agenda *agen){
    system ("cls");
    int i;
    printf("\n      %CAGENDA%C\n",207,207);
    if (cont == 0){
        printf("\n\t[!] ERROR: No hay contactos registrados [!]\n");
    }
    else{
        for(i=0;i<cont;i++){
            printf("\n%c%c%c%c Persona registrada #d
%c%c%c%c",205,205,205,205,i+1,205,205,205,205);
            printf("\nNombre completo: %s %s\n",(agen+i)->nombre,(agen+i)->apellido);
            printf("Número Teléfónico: %s\n",163,162,(agen+i)->telefono);
            printf("Correo electrónico: %s\n",162,(agen+i)->correo);
            printf("Fecha de Nacimiento: %02d/%02d/%04d\n",(agen+i)-
>fecha.dia,(agen+i)->fecha.mes,(agen+i)->fecha.anyo);
        }
    }
}

void final(char *op2,int op){
    int error;
    do{
        error = 0;
        if (op == 5){
```

```
        *op2 = 'N';
    }
    else{
        printf("\n%cDesea realizar otra operaci%cn dentro de la agenda? (s/n) %c",168,162,175);
        scanf("%s",op2);
    }
    if (*op2 == 's' || *op2 == 'S'){
        system("cls");
    }
    else{
        if (*op2 == 'n' || *op2 == 'N'){
            system("cls");
            printf("\n\n\tGRACIAS POR USAR NUESTROS SERVICIOS");
            printf("\n\n\t    Equipo 8\t-\tBUAP");
        }
        else{
            printf("\n[!] ERROR: Ingresa una opci%cn valida [!]",162);
            error = 1;
        }
    }
} while (error!=0);
}
```