



Universidad Autónoma de Baja California
Facultad de Ingeniería Arquitectura y Diseño



Ingeniero en Software y Tecnologías Emergentes.

Programación estructurada

ARCHIVOS BINARIOS

Actividad 13

Danna Guadalupe Sandez Islas

373080

1.- AGREGAR (AUTOM 100 REGISTROS)

Código:

```
case 1: /* agregar empleados al registro */
    if ((i + 100) <= N)
    {
        for (int j = 0; j < 100; j++)
        {
            empleado = gen_per();
            while (busq_opt(registro, i, empleado.key, ord) != -1)
            {
                empleado.key = numAleatorio(300000, 399999);
            }
            registro[i++] = empleado;
        }
        printf("100 empleados agregados exitosamente!\n");
    }
    else
    {
        printf("Registro lleno\n");
    }
    ord = 0;
```

Salida:

```
100 empleados agregados exitosamente!
Presione una tecla para continuar . . .
```

2.- EDITAR REGISTRO

Código:

```
void editar(Tdatos registro[], int i, int flag, char nom_bin[])
{
    int mat, op, op2;
    mat = validar("Ingrese la matricula a editar: ", 300000, 399999);
    op = busq_opt(registro, i, mat, flag);
    system("CLS");
    if (op != -1)
    {
        do
        {
            printf("1-MATRICULA: %ld\n2-NOMBRE: %s\n3-APELLIDO PATERNO: %s\n4-APELLIDO MATERNO: %s\n5-EDAD: %d\n6-SEXO: %s\n0-SALIR\n", registro[i].mat, registro[i].nom, registro[i].apaterno, registro[i].amaterno, registro[i].edad, registro[i].sexo, registro[i].salir);
            op2 = validar("Que campo desea editar: ", 0, 6);
            system("CLS");
            cambiar(op2, registro, op, flag);
            system("CLS");
        } while (op2 != 0);
        act_arch(registro, i, nom_bin);
    }
    else
    {
        printf("Matricula no encontrada\n");
    }
}
```

Salida:

```
1-MATRICULA: 322859
2-NOMBRE: CLARA
3-APELLIDO PATERNO: MEZA
4-APELLIDO MATERNO: RAMIREZ
5-EDAD: 30
6-SEXO: MUJER
0-SALIR
Que campo desea editar: █
```

```
1-MATRICULA: 373080
2-NOMBRE: DANNA
3-APELLIDO PATERNO: SANDEZ
4-APELLIDO MATERNO: ISLAS
5-EDAD: 19
6-SEXO: MUJER
0-SALIR
Que campo desea editar: █
```

3.- ELIMINAR REGISTRO (lógico)

Código:

```
void eliminar(Tdatos registro[], int i, int flag, char nom_arch[20])
{
    int mat, enc, op;
    mat = validar("Matricula a buscar: ", 300000, 399999);
    system("CLS");
    enc = busq_opt(registro, i, mat, flag);
    if (enc != -1)
    {
        if (registro[enc].status == 1)
        {
            printf("Desea eliminar a %ld?\n1-Si\n2-No\n", registro[enc].key);
            op = validar("Ingrese una opcion: ", 1, 2);
            system("CLS");
            if (op == 1)
            {
                registro[enc].status = 0;
                printf("Matricula eliminada exitosamente!\n");
                system("CLS");
                act_arch(registro, i, nom_arch);
            }
            else
            {
                printf("Matricula no eliminada\n");
            }
        }
        else
        {
            printf("Matricula inactiva\n");
        }
    }
    else
    {
        printf("La matricula que ingreso no existe\n");
    }
}
```

Salida:

```
Desea eliminar a 373080?
1-Si
2-No
Ingrese una opcion: S
```

4.- BUSCAR

Código:

```
void buscar(tdatos registro[], int i, int flag)
{
    int op, mat;
    mat = validar("Ingrese la matricula del empleado a buscar: ", 300000, 399999);
    system("CLS");
    op = busq_opt(registro, i, mat, flag);
    if (op != -1)
    {
        printf("MATRICULA: %ld\nNOMBRE: %s\nAPELLIDO PATERNO: %s\nAPELLIDO MATERNO: %s\nEDAD: %d\nSEXO: %s\n", registro[op].key, registro[op].nom, registro[op].ap_pat, registro[op].ap_mat, registro[op].edad, registro[op].sexo);
    }
    else
    {
        printf("Empleado no encontrado\n");
    }
}
```

Salida:

```
MATRICULA: 330133
NOMBRE: ELENA
APELLIDO PATERNO: TORRES
APELLIDO MATERNO: FERNANDEZ
EDAD: 27
SEXO: MUJER
Presione una tecla para continuar . . .
```

5.- ORDENAR

Código:

```
case 5: /* ordenar registros */
    if (ord == 1)
    {
        printf("El registro ya se encuentra ordenado\n");
    }
    else
    {
        ord = ordOpt(registro, i);
        printf("Ordenado\n");
        system("PAUSE");
        system("CLS");
        act_arch(registro, i, nom_bin);
    }
    break;
```

```
int ordOpt(Tdatos registro[], int n)
{
    if (n <= 200)
    {
        burbuja(registro, n);
    }
    else
    {
        quickSort(registro, n);
    }

    return 1;
}
```

```

int burbuja(Tdatos registro[], int n)
{
    int i, j;
    Tdatos temp;
    for (i = 0; i < n; i++)
    {
        for (j = i + 1; j < n; j++)
        {
            if (registro[j].key < registro[i].key)
            {
                temp = registro[i];
                registro[i] = registro[j];
                registro[j] = temp;
            }
        }
    }
    return 1;
}

int quickSort(Tdatos vect[], int n)
{
    qs(vect, 0, n - 1);

    return 1;
}

```

Salida:

```

Ordenado
Presione una tecla para continuar . . . █

```

6- IMPRIMIR

Código:

```
void imprimir(Tdatos vect[], int n, int status)
{
    int activos, i;

    printf("-----\n");
    printf(" No | MATRICULA | NOMBRE | APELLIDO P. | APELLIDO MAT. | EDAD | SEXO \n");
    printf("-----\n");
    for (i = 0, activos = 0; i < n; i++)
    {
        if (vect[i].status == status)
        {
            printf("%4d.- %6ld %10s %10s %10s %2d %-7s\n", activos + 1, vect[i].key, vect[i].nom, vect[i].ap_pat, vect[i].ap_mat, vect[i].edad, vect[i].sexo);
            activos++;
        }
    }
}
```

Salida:

No	MATRICULA	NOMBRE	APELLIDO P.	APELLIDO MAT.	EDAD	SEXO
1.-	300011	VICTORIA	LOPEZ	SAAVEDRA	37	MUJER
2.-	300613	ISABEL	SANDEZ	SANCHEZ	36	MUJER
3.-	300787	JAVIER	RODRIGUEZ	RUIZ	24	HOMBRE
4.-	300979	SOFIA	ORTEGA	VARGAS	39	MUJER
5.-	301376	CARMEN	TORRES	MEZA	30	MUJER
6.-	301512	FELIPE	ORTEGA	CASTILLO	28	HOMBRE
7.-	301845	ANDRES	SOLANO	GARCIA	33	HOMBRE
8.-	303527	BEATRIZ	PEREZ	TORRES	30	MUJER
9.-	303558	ELENA	LOPEZ	PEREZ	31	MUJER
10.-	304195	LUISA	MENDOZA	SANDEZ	28	MUJER
11.-	304328	PATRICIA	LOPEZ	TORRES	18	MUJER
12.-	304458	BEATRIZ	SANDEZ	RAMIREZ	25	MUJER
13.-	305234	JOSE	TORRES	VARGAS	19	MUJER
14.-	305502	GABRIELA	RAMIREZ	GARCIA	26	MUJER
15.-	305535	PATRICIA	RUIZ	ROJAS	32	MUJER
16.-	306376	MIGUEL	RODRIGUEZ	LOPEZ	18	HOMBRE
17.-	306596	PEDRO	MARTINEZ	DIAZ	24	HOMBRE
18.-	306974	ANDRES	SANCHEZ	LOPEZ	22	HOMBRE
19.-	307851	JOSE	GONZALEZ	LOPEZ	38	HOMBRE
20.-	308179	PATRICIA	PEREZ	ROJAS	39	MUJER
21.-	309073	ALEJANDRO	SANDEZ	MORALES	22	HOMBRE
22.-	309258	ALEJANDRO	TORRES	MORALES	20	HOMBRE
23.-	309497	LAURA	VARGAS	VARGAS	33	MUJER
24.-	310010	SOFIA	ORTEGA	MORALES	20	MUJER
25.-	310091	FERNANDO	RAMIREZ	FLORES	28	HOMBRE
26.-	310881	JAVIER	VARGAS	MARTINEZ	29	HOMBRE
27.-	311745	ELENA	TORRES	MORALES	37	MUJER
28.-	311862	VICTORIA	RAMIREZ	TORRES	20	MUJER
29.-	312227	FELIPE	ROJAS	GUTIERREZ	38	HOMBRE
30.-	312423	SEBASTIAN	MENDOZA	GUTIERREZ	26	HOMBRE
31.-	312699	CLARA	RODRIGUEZ	PEREZ	32	MUJER
32.-	312859	JOSE	ROJAS	TORRES	39	HOMBRE
33.-	312871	VICTORIA	MORALES	LOPEZ	21	MUJER
34.-	313317	CARLOS	GARCIA	MEZA	31	HOMBRE
35.-	313953	ANDRES	FERNANDEZ	VARGAS	30	HOMBRE
36.-	314488	ISABEL	FLORES	PEREZ	32	MUJER
37.-	314489	ROBERTO	PEREZ	RODRIGUEZ	28	HOMBRE
38.-	314586	MANUEL	MENDOZA	GONZALEZ	33	HOMBRE
39.-	314767	ELENA	GONZALEZ	PEREZ	24	MUJER
40.-	315033	ELENA	PEREZ	GARCIA	37	MUJER
41.-	315084	ANTONIO	SANCHEZ	SANCHEZ	28	HOMBRE
42.-	315752	LAURA	GONZALEZ	TORRES	19	MUJER

7.- GENERAR ARCHIVO TEXTO

Código:

```
case 7: /* generar archivo de texto */
    gen_nom_arch(nom_arch);
    gen_arch(nom_arch, registro, i); // empleados activos
    printf("Tarea realizada exitosamente!\n");
    break;
```

```
void gen_nom_arch(char nom_arch[])
{
    do
    {
        system("CLS");
        printf("Nombre del archivo: ");
        fflush(stdin);
        gets(nom_arch);
    } while (strcmp(nom_arch, "\n") == 0);
}
```

```
void gen_arch(char nomArchivo[], Tdatos vect[], int n)
{
    int i, cont = 0;
    char temp[30];
    FILE *fa;

    strcpy(temp, nomArchivo);
    strcat(temp, "_activos.txt");

    fa = fopen(temp, "w");

    if (fa)
    {
        fprintf(fa, "-----\n");
        fprintf(fa, " No | MATRICULA | NOMBRE | APELLIDO P. | APELLIDO MAT. | EDAD | SEXO \n");
        fprintf(fa, "-----\n");

        for (i = 0; i < n; i++)
        {
            if (vect[i].status == 1)
            {
                fprintf(fa, "%4d.- %6ld %10s %10s %10s %2d %7s", cont, vect[i].key, vect[i].nom, vect[i].ap_pat
                fprintf(fa, "\n");
                cont++;
            }
        }
    }
    else
    {
        printf("Al parecer ha ocurrido un error al generar el archivo\n");
    }
    fclose(fa);
}
```

Salida:

```
Nombre del archivo: empleados
Tarea realizada exitosamente!
Presione una tecla para continuar . . . █
```

main.cpp							
empleados_activos.txt X							
options_background.png							
FondoMenu.gif							
LEVEL_.mp3							
Danna_Sandez_Programacion_Estructurada > Actividad_13 > output > empleados_activos.txt							
1							
2	No	MATRICULA	NOMBRE	APELLIDO P.	APELLIDO MAT.	EDAD	SEXO
3							
4	0.-	300011	VICTORIA	LOPEZ	SAAVEDRA	37	MUJER
5	1.-	300613	ISABEL	SANDEZ	SANCHEZ	36	MUJER
6	2.-	300787	JAVIER	RODRIGUEZ	RUIZ	24	HOMBRE
7	3.-	300979	SOFIA	ORTEGA	VARGAS	39	MUJER
8	4.-	301376	CARMEN	TORRES	MEZA	30	MUJER
9	5.-	301512	FELIPE	ORTEGA	CASTILLO	28	HOMBRE
10	6.-	301845	ANDRES	SOLANO	GARCIA	33	HOMBRE
11	7.-	303527	BEATRIZ	PEREZ	TORRES	30	MUJER
12	8.-	303558	ELENA	LOPEZ	PEREZ	31	MUJER
13	9.-	304195	LUISA	MENDOZA	SANDEZ	28	MUJER
14	10.-	304328	PATRICIA	LOPEZ	TORRES	18	MUJER
15	11.-	304458	BEATRIZ	SANDEZ	RAMIREZ	25	MUJER
16	12.-	305234	JOSE	TORRES	VARGAS	19	MUJER
17	13.-	305502	GABRIELA	RAMIREZ	GARCIA	26	MUJER
18	14.-	305535	PATRICIA	RUIZ	ROJAS	32	MUJER
19	15.-	306376	MIGUEL	RODRIGUEZ	LOPEZ	18	HOMBRE
20	16.-	306596	PEDRO	MARTINEZ	DIAZ	24	HOMBRE
21	17.-	306974	ANDRES	SANCHEZ	LOPEZ	22	HOMBRE
22	18.-	307851	JOSE	GONZALEZ	LOPEZ	38	HOMBRE
23	19.-	308179	PATRICIA	PEREZ	ROJAS	39	MUJER
24	20.-	309073	ALEJANDRO	SANDEZ	MORALES	22	HOMBRE
25	21.-	309258	ALEJANDRO	TORRES	MORALES	20	HOMBRE
26	22.-	309497	LAURA	VARGAS	VARGAS	33	MUJER
27	23.-	310010	SOFIA	ORTEGA	MORALES	20	MUJER
28	24.-	310091	FERNANDO	RAMIREZ	FLORES	28	HOMBRE
29	25.-	310881	JAVIER	VARGAS	MARTINEZ	29	HOMBRE
30	26.-	311745	ELENA	TORRES	MORALES	37	MUJER
31	27.-	311862	VICTORIA	RAMIREZ	TORRES	20	MUJER
32	28.-	312227	FELIPE	ROJAS	GUTIERREZ	38	HOMBRE
33	29.-	312423	SEBASTIAN	MENDOZA	GUTIERREZ	26	HOMBRE
34	30.-	312699	CLARA	RODRIGUEZ	PEREZ	32	MUJER
35	31.-	312859	JOSE	ROJAS	TORRES	39	HOMBRE
36	32.-	312871	VICTORIA	MORALES	LOPEZ	21	MUJER
37	33.-	313317	CARLOS	GARCIA	MEZA	31	HOMBRE
38	34.-	313953	ANDRES	FERNANDEZ	VARGAS	30	HOMBRE

8.- VER ARCHIVO TEXTO

Código:

```
void mostrar_arch(tdatos vect[], int n, char nom_arch[])
{
    char temp[20];
    int cont = 0;
    gen_nom_arch(temp);
    strcat(temp, "_activos.txt");
    if (strcmp(temp, nom_arch))
    {
        printf("-----\n");
        printf(" No | MATRICULA | NOMBRE | APELLIDO P. | APELLIDO MAT. | EDAD | SEXO \n");
        printf("-----\n");
        for (int i = 0; i < n; i++)
        {
            if (vect[i].status == 1)
            {
                printf("%4d.- %6ld %-10s %-10s %-10s %2d %-7s", cont, vect[i].key, vect[i].nom, vect[i].nom, vect[i].nom, vect[i].edad, vect[i].sexo);
                printf("\n");
                cont++;
            }
        }
    }
    else
    {
        printf("El archivo que buscas no existe\n");
    }
}
```

Salida:

Nombre del archivo: empleados						
No	MATRICULA	NOMBRE	APELLIDO P.	APELLIDO MAT.	EDAD	SEXO
0.-	300011	VICTORIA	LOPEZ	SAAVEDRA	37	MUJER
1.-	300613	ISABEL	SANDEZ	SANCHEZ	36	MUJER
2.-	300787	JAVIER	RODRIGUEZ	RUIZ	24	HOMBRE
3.-	300979	SOFIA	ORTEGA	VARGAS	39	MUJER
4.-	301376	CARMEN	TORRES	MEZA	30	MUJER
5.-	301512	FELIPE	ORTEGA	CASTILLO	28	HOMBRE
6.-	301845	ANDRES	SOLANO	GARCIA	33	HOMBRE
7.-	303527	BEATRIZ	PEREZ	TORRES	30	MUJER
8.-	303558	ELENA	LOPEZ	PEREZ	31	MUJER
9.-	304195	LUISA	MENDOZA	SANDEZ	28	MUJER
10.-	304328	PATRICIA	LOPEZ	TORRES	18	MUJER
11.-	304458	BEATRIZ	SANDEZ	RAMIREZ	25	MUJER
12.-	305234	JOSE	TORRES	VARGAS	19	MUJER
13.-	305502	GABRIELA	RAMIREZ	GARCIA	26	MUJER
14.-	305535	PATRICIA	RUIZ	ROJAS	32	MUJER
15.-	306376	MIGUEL	RODRIGUEZ	LOPEZ	18	HOMBRE
16.-	306596	PEDRO	MARTINEZ	DIAZ	24	HOMBRE
17.-	306974	ANDRES	SANCHEZ	LOPEZ	22	HOMBRE
18.-	307851	JOSE	GONZALEZ	LOPEZ	38	HOMBRE
19.-	308179	PATRICIA	PEREZ	ROJAS	39	MUJER
20.-	309073	ALEJANDRO	SANDEZ	MORALES	22	HOMBRE
21.-	309258	ALEJANDRO	TORRES	MORALES	20	HOMBRE
22.-	309497	LAURA	VARGAS	VARGAS	33	MUJER
23.-	310010	SOFIA	ORTEGA	MORALES	20	MUJER
24.-	310091	FERNANDO	RAMIREZ	FLORES	28	HOMBRE
25.-	310881	JAVIER	VARGAS	MARTINEZ	29	HOMBRE
26.-	311745	ELENA	TORRES	MORALES	37	MUJER
27.-	311862	VICTORIA	RAMIREZ	TORRES	20	MUJER
28.-	312227	FELIPE	ROJAS	GUTIERREZ	38	HOMBRE
29.-	312423	SEBASTIAN	MENDOZA	GUTIERREZ	26	HOMBRE
30.-	312699	CLARA	RODRIGUEZ	PEREZ	32	MUJER
31.-	312859	JOSE	ROJAS	TORRES	39	HOMBRE
32.-	312871	VICTORIA	MORALES	LOPEZ	21	MUJER
33.-	313317	CARLOS	GARCIA	MEZA	31	HOMBRE
34.-	313953	ANDRES	FERNANDEZ	VARGAS	30	HOMBRE
35.-	314488	ISABEL	FLORES	PEREZ	32	MUJER
36.-	314489	ROBERTO	PEREZ	RODRIGUEZ	28	HOMBRE

9.- CREAR ARCH BINARIO

Código:

```
case 9: /* crear archivo binario */
    gen_nom_arch(nom_bin);
    gen_arch_bin(registro, i, nom_bin);
    respaldo(nom_bin);
    printf("Tarea realizada exitosamente!\n");
    break;
```

```
int gen_arch_bin(Tdatos registro[], int n, char nom[])
{
    Tdatos reg;
    FILE *fa;
    int i;
    char temp[20];
    strcpy(temp, nom);
    strcat(temp, ".dll");
    fa = fopen(temp, "ab");
    if (fa == NULL)
    {
        printf("No hay registros por guardar\n");
        return 0;
    }
    else
    {
        if (fa)
        {
            for (i = 0; i < n; i++)
            {
                reg = registro[i];
                fwrite(&reg, sizeof(Tdatos), 1, fa);
            }
            fclose(fa);
        }
    }

    return 1;
}
```

```

void respaldo(char nom[])
{
    FILE *fa;
    FILE *resp;
    Tdatos reg;
    char nom2[20];
    strcpy(nom2, nom);
    strcat(nom2, ".dll");
    fa = fopen(nom2, "rb");
    if (fa)
    {
        strcat(nom, ".tmp");
        resp = fopen(nom, "wb");
        while (fread(&reg, sizeof(Tdatos), 1, fa))
        {
            fwrite(&reg, sizeof(Tdatos), 1, resp);
        }
        fclose(fa);
        fclose(resp);
    }
}

```

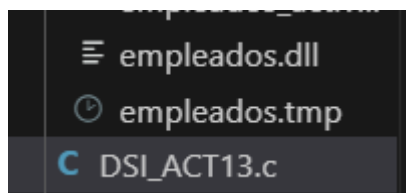
Tdatos registro[1]

Salida:

```

Nombre del archivo: empleados
Tarea realizada exitosamente!
Presione una tecla para continuar . . . █

```



10.- CARGAR ARCH BINARIO

Código:

```
case 10: /* cargar archivo binario */
    if (carg)
    {
        printf("El archivo ya se encuentra cargado\n");
    }
    else
    {
        carg = cargar_bin(registro, &i, N, nom_bin);
    }
    break;
```

```
int cargar_bin(Tdatos registro[], int *i, int max, char nom[])
{
    Tdatos reg;
    FILE *fa;
    FILE *crg;
    char nom2[20];
    fa = fopen(nom, "rb");
    if (fa)
    {
        strcpy(nom2, "datos.dll");
        crg = fopen(nom2, "wb");
        while (fread(&reg, sizeof(Tdatos), 1, crg))
        {
            if ((*i) <= max)
            {
                registro[(*i)++] = reg;
            }
        }
        fclose(fa);
        fclose(crg);
        printf("Archivo cargado exitosamente\n");
        return 1;
    }
    else
    {
        printf("Al parecer ha ocurrido un error\n");
        return 0;
    }
}
```

Salida:

```
Archivo cargado exitosamente  
Presione una tecla para continuar . . .
```

```
≡ datos.dll
```

11.- MOSTRAR ELIMINADOS

Código:

```
case 11: /*mostrar borrados */
    imprimir(registro, i, 0);
    break;
```

```
void imprimir(Tdatos vect[], int n, int status)
{
    int activos, i;

    printf("-----\n");
    printf(" No  | MATRICULA | NOMBRE      | APELLIDO P.  | APELLIDO MAT.  | EDAD  | SEXO \n");
    printf("-----\n");
    for (i = 0, activos = 0; i < n; i++)
    {
        if (vect[i].status == status)
        {
            printf("%4d.- %6ld    %-10s    %-10s    %-10s    %2d    %-7s\n", activos + 1, vect[i].key, vect[i].nom, vect[i].ap_pat, vect[i].ap_mat, vect[i].edad, vect[i].sexo);
            activos++;
        }
    }
}
```

Salida:

```
-----
 No  | MATRICULA | NOMBRE      | APELLIDO P.  | APELLIDO MAT.  | EDAD  | SEXO
-----
 1.-  373080    DANNA      SANDEZ        ISLAS         19     MUJER
Presione una tecla para continuar . . .
```