

UNIVERSIDAD AUTÓNOMA DE BAJA CALIFORNIA



Ingeniería en Computación

Anexos actividad 13

Programación Estructurada

ALUMNO: Arredondo Urbalejo Isai Alexis

MATRÍCULA: 368747

GRUPO: 932

PROFESOR: Pedro Nunez Yepiz



Programación Estructurada



```
File Edit Selection View Go Run ... Programacion
EXPLORER
PROGRAMACION
> .vscode
> Actividad2
> Actividad3
> Actividad4
> Actividad5
> Actividad6
> Actividad7
> Actividad8
> Actividad9
> Actividad92
> Actividad93
> Actividad10
> Actividad11
> Actividad12
> Actividad13
> Actividad13py
> Actividad14
> alexis.h
> Ejercicio13.1.cpp
> Ejemplo
> Examen
> Juego
> Juego2
> Marcelo
> Mates
> revision
> Software
OUTLINE
TIMELINE
Ln 17, Col 12 Spaces: 4 UTF-8 CRLF C++ Win32
```

```
Actividad14 > Ejercicio13.1.cpp > talum
1 #include <stdio.h>
2 #include <stdlib.h>
3 #include <string.h>
4 #include <ctype.h>
5 #include <time.h>
6 #include <windows.h>
7 #include "alexis.h"
8 #define N 2500
9 typedef struct talum{
10     long matricula;
11     char nombre[20];
12     char ApPat[20];
13     char ApMat[20];
14     int edad;
15     char sexo[20];
16     int status;
17 }talum;
18
19 int msges();
20 void menu();
21 talum agregar(talum vect[],int n);
22 int validar_matri(int matri, talum vector[], int n);
23 void imprimir(talum vect[], int n);
24 void generartxt(talum vect[],int n,char nombrearchivo[]);
25 int main()
26 {
27     srand(time(NULL));
28     menu();
29     return 0;
30 }
31
32 int msges()
33 {
34     int op,menu;
35     system("CLS");
36     printf("-----\n");
37     printf("  H E N U \n");
38     printf("1.- AGREGAR AUTOMATICO \n");
39     printf("2.- EDITAR REGISTRO \n");
```

```
File Edit Selection View Go Run ... Programacion
EXPLORER
PROGRAMACION
> .vscode
> Actividad2
> Actividad3
> Actividad4
> Actividad5
> Actividad6
> Actividad7
> Actividad8
> Actividad9
> Actividad92
> Actividad93
> Actividad10
> Actividad11
> Actividad12
> Actividad13
> Actividad13py
> Actividad14
> alexis.h
> Ejercicio13.1.cpp
> Ejemplo
> Examen
> Juego
> Juego2
> Marcelo
> Mates
> revision
> Software
OUTLINE
TIMELINE
Ln 17, Col 12 Spaces: 4 UTF-8 CRLF C++ Win32
```

```
Actividad14 > Ejercicio13.1.cpp > talum
40 printf("3.- ELIMINAR REGISTRO \n");
41 printf("4.- BUSCAR \n");
42 printf("5.- ORDENAR \n");
43 printf("6.- IMPRIMIR \n");
44 printf("7.- GENERAR ARCHIVO TEXTO \n");
45 printf("8.- VER ARCHIVO TEXTO \n");
46 printf("9.- CREAR ARCH BINARIO \n");
47 printf("10.- CARGAR ARCH BINARIO \n");
48 printf("0.- SALIR \n");
49 printf("-----\n");
50 op=validanum_int(0,10,"ELIGE UNA OPCION:", "ERROR");
51 return op;
52 }
53
54 void menu()
55 {
56     int i,op,total,tam=0,j,k=0,l=0;
57     char nom_arch[20] = "Alumnos";
58     char nom_arch_bin[20] = "archi";
59     char nombreachivo[20];
60     bool cargado_bin = false;
61     talum vect[N];
62     talum reg;
63     do{
64         l = k+100;
65         op=msges();
66         switch (op)
67         {
68             case 1:
69                 if(k < N-100)
70                 {
71                     for(i=k; i<l; i++)
72                     {
73                         vect[i]=agregar(vect,k);
74                         k++;
75                     }
76                     break;
77                 case 2:
```



Programación Estructurada

```
File Edit Selection View Go Run ... Programación
EXPLORER
PROGRAMACION
Actividad14 > Ejercicio13.1.cpp
79 imprimir(vect,k);
80 break;
81 }
82 while (op != 0);
83 generatxt(vect,k,nom_arch);
84 system("PAUSE");
85 }
86 int cargar_archivo(talum vect[],int n)
87 {
88     int i;
89     i = n;
90     talum reg;
91     char basura[5];
92     FILE *arch;
93     arch = fopen("datos.txt","r");
94     if(arch)
95     {
96         while(!feof(arch))
97         {
98             fscanf(arch,"%s%d%s%s%s", &basura, &reg.matricula, &reg.nombre, &reg.ApPat, &reg.ApMat, &reg.edad, &reg.sexo);
99             vect[i++] = reg;
100             if(i == N)
101             {
102                 return i;
103             }
104             fclose(arch);
105         }
106         return i;
107     }
108     return i;
109 }
110 talum agregar(talum vect[],int n)
111 {
112     talum reg;
113     char nombres[20][20]={"MARCELO","ALEXIS","JULIO","EMILIANO","SERGIO","ABRAHAM","ROBERTO","JACOB","JOSUE","PABLO","ALMAHIA"};
114     char mujeres[20][20]={"ANA","SAWANTHA","KARLA","MANCY","ALEXA","DENISSE","DANIELA","VIVIANA","HANNA","LEWY","JULISA"};
115     char ApPat[20][20]={"MARQUEZ","RUIZ","PEREZ","ARREDONDO","ZAVALA","ANDRADE","ROJAS","LUNA","HERNANDEZ","GARCIA","SHADE","SIQUEIROS"};
116     char ApMat[20][20]={"MARQUEZ","RUIZ","PEREZ","ARREDONDO","ZAVALA","ANDRADE","ROJAS","LUNA","HERNANDEZ","GARCIA","SHADE","SIQUEIROS"};
117 }
```

```
File Edit Selection View Go Run ... Programación
EXPLORER
PROGRAMACION
Actividad14 > Ejercicio13.1.cpp
118 char mujeres[20][20]={"MARQUEZ","RUIZ","PEREZ","ARREDONDO","ZAVALA","ANDRADE","ROJAS","LUNA","HERNANDEZ","GARCIA","SHADE","SIQUEIROS"};
119 char ApPat[20][20]={"MARQUEZ","RUIZ","PEREZ","ARREDONDO","ZAVALA","ANDRADE","ROJAS","LUNA","HERNANDEZ","GARCIA","SHADE","SIQUEIROS"};
120 char ApMat[20][20]={"MARQUEZ","RUIZ","PEREZ","ARREDONDO","ZAVALA","ANDRADE","ROJAS","LUNA","HERNANDEZ","GARCIA","SHADE","SIQUEIROS"};
121 int sexo,status=1;
122 long r13=300000, rf3=399999, rango3,matri;
123 strcpy(reg.ApPat,ApPat[rand()%(N12)]);
124 strcpy(reg.ApMat,ApMat[rand()%(N12)]);
125 sexo=(rand()%(2))+1;
126 if (sexo==1)
127 {
128     strcpy(reg.sexo,"HOMBRE");
129     strcpy(reg.nombre,nombres[rand()%(N11)]);
130 }
131 else
132 {
133     strcpy(reg.sexo,"MUJER");
134     strcpy(reg.nombre,mujeres[rand()%(N11)]);
135 }
136 reg.edad=(rand()%(N10))+12;
137 rango3 = ( rf3 - r13 ) + 1;
138 do{
139     matri = (rand()%(rango3) + r13);
140 }while(validar_matri(matri,vect,n) != -1);
141 reg.matricula = matri;
142 reg.status=status;
143 return reg;
144 }
145 void imprimir(talum vect[], int n)
146 {
147     system("CLS");
148     int i;
149     printf(" MATRICULA NOMBRE APELLIDO PAT APELLIDO MAT EDAD SEXO \n");
150     for(i=0; i<n; i++)
151     {
152         if (vect[i].status==1)
153         {
154             printf("%5d %10s %12s %12s %5d %10s \n", vect[i].matricula, vect[i].nombre, vect[i].ApPat, vect[i].ApMat, vect[i].edad, vect[i].sexo);
155         }
156     }
157 }
```



Programación Estructurada



```
153     if (vect[i].status==1)
154     {
155         printf(" %d %10s %12s %12s %d %10s \n",vect[i].matricula,vect[i].nombre,vect[i].ApPat,vect[i].ApMat,vect[i].edad,vect[i].sexo);
156     }
157 }
158 system("PAUSE");
159
160
161 void generartxt(talum vect[],int n,char nombreachivo[])
162 {
163     int i;
164     char nombre_arch[24];
165     strcpy(nombre_arch,nombreachivo);
166     strcat(nombre_arch, ".txt");
167     FILE *arch;
168     arch = fopen(nombre_arch,"w");
169     fprintf(arch,"----- \n");
170     fprintf(arch," No | MATRICULA | NOMBRE | APELLIDO P. | APELLIDO MAT. | EDAD | SEXO | \n");
171     fprintf(arch,"----- \n");
172     if(arch)
173     {
174         for(i=0; i<n; i++)
175         {
176             if (vect[i].status==1)
177             {
178                 fprintf(arch,"%d %10s %12s %12s %d %10s \n",i,vect[i].matricula,vect[i].nombre,vect[i].ApPat,vect[i].ApMat,vect[i].edad,vect[i].sexo);
179             }
180         }
181         fclose(arch);
182     }
183 }
```