

**UNIVERSIDAD AUTONOMA DE BAJA CALIFORNIA**  
**FACULTAD DE INGENIERIA ARQUITECTURA Y DISEÑO**



**PROGRAMACION ESTRUCTURADA**

**ANEXO 8**

**GABRIEL TEJEDA CHAVEZ**

**M:347723**



**ENSENADA A 24 DE MARZO DE 2024**

ACTIVIDAD 8 > GTC\_ACT08\_01.cpp > msges()

```
1  // GABRIEL TEJEDA CHAVEZ
2  // PROGRAMA QUE CREA E IMPRIME ARREGLOS NUMERICOS
3  // 19/03/2024
4
5  #include <stdio.h>
6  #include <stdlib.h>
7  #include <time.h>
8  #define N 10
9
10 int msges();
11 void menu();
12 void llenar_vector(int vect[], int m);
13 void vector_num_aleat(int vect[], int m);
14 void vector_datos_arreglo(int vect1[], int vect2[], int m);
15 void imprimir_vector1(int vect[], int m, char nombre[]);
16 void imprimir_vector2(int vect[], int m, char nombre[]);
17 void imprimir_vector3(int vect[], int vect2[], int vect3[], int m, char nombre[]);
18 void llenar_matriz(int vect1[], int vect2[], int m);
19 void imprimir_matriz(int vect1[], int vect2[], int m);
20 int buscador(int vect2[], int num, int m);
21
22 int main()
23 {
24     menu();
25     system("CLS");
26     return 0;
27 }
28
29 int msges()
30 {
```

```

30 {
31     int op;
32     system("CLS");
33     printf("Men%c\n", 163);
34     printf("1.- Llenar vector 1 (manualmente) \n");
35     printf("2.- Llenar vector 2 (aleatoriamente) \n");
36     printf("3.- Llenar vector 3 (vector 1 y vector 2) \n");
37     printf("4.- Imprimir vectores \n");
38     printf("5.- Llena matriz 4*4 \n");
39     printf("6.- Imprimir matriz \n");
40     printf("0.- Salir \n");
41     printf("Escoge una opci%cn: ", 162);
42     scanf("%d", &op);
43     return op;
44 }
45
46 void menu()
47 {
48     int op, vect1[N], vect2[N], vect3[21], matriz[2][N];
49     do
50     {
51         op = msges();
52         switch (op)
53         {
54             case 1:
55                 Llenar_vector(vect1, N);
56                 break;
57
58             case 2:

```

```

59         vector_num_aleat(vect2, N);
60         break;
61
62     case 3:
63         vector_datos_arreglo(vect1, vect2, N);
64         break;
65
66     case 4:
67         imprimir_vector1(vect1, N, "VECTOR 1");
68         imprimir_vector2(vect2, N, "VECTOR 2");
69         imprimir_vector3(vect1, vect2, vect3, N, "VECTOR 3");
70         break;
71
72     case 5:
73         llenar_matriz(vect1, vect2, N);
74         break;
75
76     case 6:
77         imprimir_matriz(vect1, vect2, N);
78         break;
79     }
80
81     } while (op != 0);
82 }
83
84 void llenar_vector(int vect1[], int m)
85 {
86     system("CLS");
87     int i = 0;

```

```

87     int i, j;
88
89     for (i = 0; i < m; i++)
90     {
91         j = i;
92         do
93         {
94             system("CLS");
95             printf("Ingrese un numero entero que este entre el 30 y el 70 para el vector[%d]: ", i);
96             scanf("%d", &vect1[j]);
97             if (vect1[j] >= 30 && vect1[j] <= 70)
98             {
99                 j++;
100             }
101             else
102             {
103                 printf("Valor incorrecto, intentelo nuevamente\n");
104                 system("PAUSE");
105             }
106         } while (j <= i);
107     }
108 }
109
110
111 int buscador(int vect2[], int num, int m)
112 {
113     int i;
114     for (i = 0; i < m; i++)
115     {

```

```

115     {
116         if (vect2[i] == num)
117         {
118             return 1;
119         }
120     }
121     return 0;
122 }
123
124 void vector_num_aleat(int vect2[], int m)
125 {
126     system("CLS");
127     int i, j, k, repetido, num;
128     srand(time(NULL));
129
130     i = 0;
131     while (i < m)
132     {
133         num = (rand() % 20) + 1;
134         if (buscador(vect2, num, m) == 0)
135         {
136             vect2[i] = num;
137             i++;
138         }
139     }
140 }
141
142 void vector_datos_arreglo(int vect1[], int vect2[], int m)

```

```
143 {
144     system("CLS");
145
146     int vect3[20], i;
147
148     for (i = 0; i < 10; i++)
149     {
150         vect3[i] = vect1[i];
151     }
152
153     for (i = 0; i < 10; i++)
154     {
155         vect3[i + 10] = vect2[i];
156     }
157 }
158
159 void imprimir_vector1(int vect1[], int m, char nombre[])
160 {
161     int i;
162     system("CLS");
163     printf("\n\n %s \n", nombre);
164     for (i = 0; i < m; i++)
165     {
166         printf("Vect1[%d] = %d\n", i, vect1[i]);
167     }
168 }
169
170 void imprimir_vector2(int vect2[], int m, char nombre[])
171 {
```

```
172     int i;
173     printf("\n\n %s  \n", nombre);
174     for (i = 0; i < m; i++)
175     {
176         printf("Vect2[%d] = %d\n", i, vect2[i]);
177     }
178 }
179
180 void imprimir_vector3(int vect1[], int vect2[], int vect3[], int m, char nombre[])
181 {
182     int i;
183     printf("\n\n %s  \n", nombre);
184
185     for (i = 0; i < 10; i++)
186     {
187         vect3[i] = vect1[i];
188     }
189
190     for (i = 0; i < 10; i++)
191     {
192         vect3[i + 10] = vect2[i];
193     }
194
195     for (i = 0; i < 20; i++)
196     {
197         printf("Vector[%d]= %d\n", i, vect3[i]);
198     }
199     system("PAUSE");
```



```
200     }
201
202     void llenar_matriz(int vect1[], int vect2[], int m)
203     {
204         system("CLS");
205         int matriz[2][N];
206         int i, j;
207
208         for (i = 0; i < m; i++)
209         {
210             matriz[0][i] = vect1[i];
211         }
212
213         for (i = 0; i < m; i++)
214         {
215             matriz[1][i] = vect2[i];
216         }
217     }
218
219     void imprimir_matriz(int vect1[], int vect2[], int m)
220     {
221         system("CLS");
222         int matriz[2][N];
223         int i, j;
224
225         for (i = 0; i < m; i++)
226         {
227             matriz[0][i] = vect1[i];
228         }
```

```

228     }
229
230     for (i = 0; i < m; i++)
231     {
232         matriz[1][i] = vect2[i];
233     }
234
235     printf("Matriz resultante:\n");
236     for (int i = 0; i < 2; i++)
237     {
238         for (int j = 0; j < m; j++)
239         {
240             printf("%d ", matriz[i][j]);
241         }
242         printf("\n");
243     }
244     system("PAUSE");
245 }
246

```

PROBLEMS

3

OUTPUT

DEBUG CONSOLE

TERMINAL

PORTS

Menú

- 1.- Llenar vector 1 (manualmente)
- 2.- Llenar vector 2 (aleatoriamente)
- 3.- Llenar vector 3 (vector 1 y vector 2)
- 4.- Imprimir vectores
- 5.- Llena matriz 4\*4
- 6.- Imprimir matriz
- 0.- Salir

Escoge una opción: █

Ingrese un numero entero que este entre el 30 y el 70 para el vector[0]: █

Ingrese un numero entero que este entre el 30 y el 70 para el vector[0]: 20

Ingrese un numero entero que este entre el 30 y el 70 para el vector[0]: 20  
Valor incorrecto, intentelo nuevamente  
Press any key to continue . . .

VECTOR 1  
Vect1[0] = 30  
Vect1[1] = 30  
Vect1[2] = 30  
Vect1[3] = 30  
Vect1[4] = 30  
Vect1[5] = 30  
Vect1[6] = 30  
Vect1[7] = 30  
Vect1[8] = 30  
Vect1[9] = 30

VECTOR 2  
Vect2[0] = 12  
Vect2[1] = 9  
Vect2[2] = 4  
Vect2[3] = 10  
Vect2[4] = 3  
Vect2[5] = 13  
Vect2[6] = 2  
Vect2[7] = 19  
Vect2[8] = 20  
Vect2[9] = 5

```
VECTOR 3
Vector[0]= 30
Vector[1]= 30
Vector[2]= 30
Vector[3]= 30
Vector[4]= 30
Vector[5]= 30
Vector[6]= 30
Vector[7]= 30
Vector[8]= 30
Vector[9]= 30
Vector[10]= 12
Vector[11]= 9
Vector[12]= 4
Vector[13]= 10
Vector[14]= 3
Vector[15]= 13
Vector[16]= 2
Vector[17]= 19
Vector[18]= 20
Vector[19]= 5
Press any key to continue . . .
```

```
Matriz resultante:
30 30 30 30 30 30 30 30 30 30
12 9 4 10 3 13 2 19 20 5
Press any key to continue . . .
```