



Universidad Autónoma de Baja
California
Facultad de Ingeniería Arquitectura y
Diseño



Materia: Programación Estructurada

Alumno: Fabian Aguiar Sergio

Carrera: Ingeniero en computación

Fecha: 22/03/2024

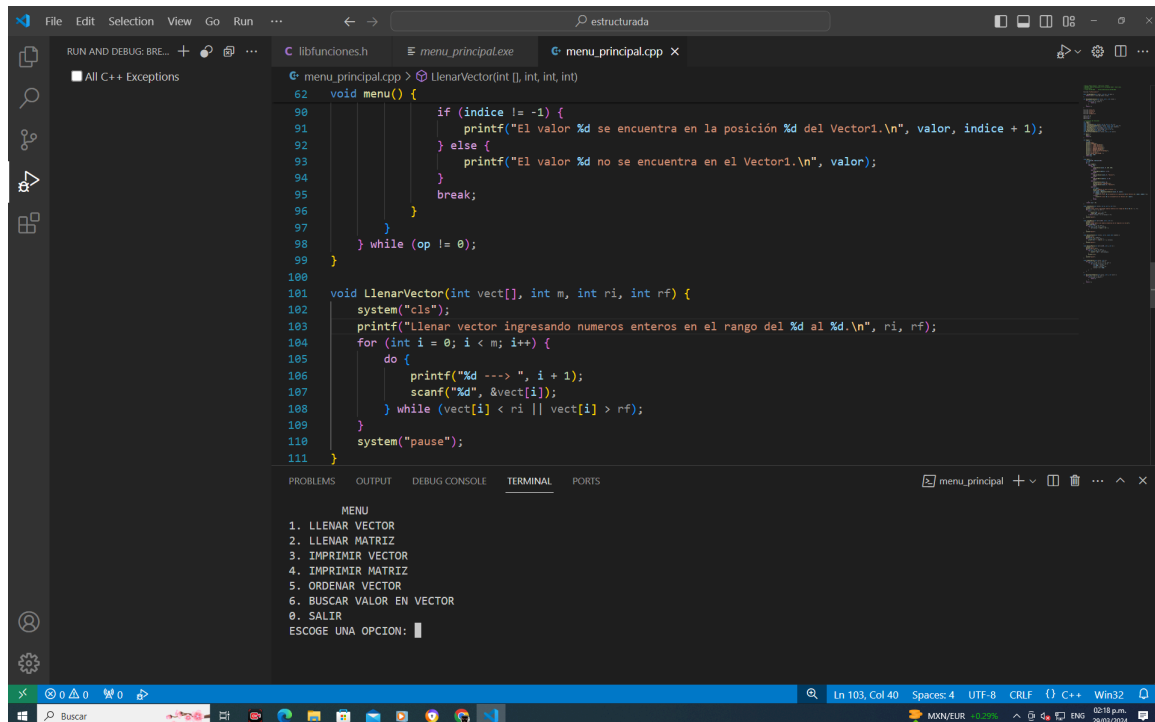
Matrícula: 374317

Maestro: Pedro Nuñez Yepiz

Práctica No. #8

ANEXOS

-1.- LLENAR VECTOR .- Llenar vector con 15 números, los números generados aleatoriamente, los números entre el rango de 100 al 200 (**no repetidos**)



```
1  menu_principal.cpp > LlenarVector(int [], int, int, int)
2
3  void menu() {
4      if (indice != -1) {
5          printf("El valor %d se encuentra en la posición %d del Vector1.\n", valor, indice + 1);
6      } else {
7          printf("El valor %d no se encuentra en el Vector1.\n", valor);
8      }
9      break;
10 }
11 } while (op != 0);
12 }
13
14 void LlenarVector(int vect[], int m, int ri, int rf) {
15     system("cls");
16     printf("Llenar vector ingresando numeros enteros en el rango del %d al %d.\n", ri, rf);
17     for (int i = 0; i < m; i++) {
18         do {
19             printf("%d ---> ", i + 1);
20             scanf("%d", &vect[i]);
21             while (vect[i] < ri || vect[i] > rf);
22         } while (true);
23     }
24     system("pause");
25 }
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

menu_principal

MENU

1. LLENAR VECTOR

2. LLENAR MATRIZ

3. IMPRIMIR VECTOR

4. IMPRIMIR MATRIZ

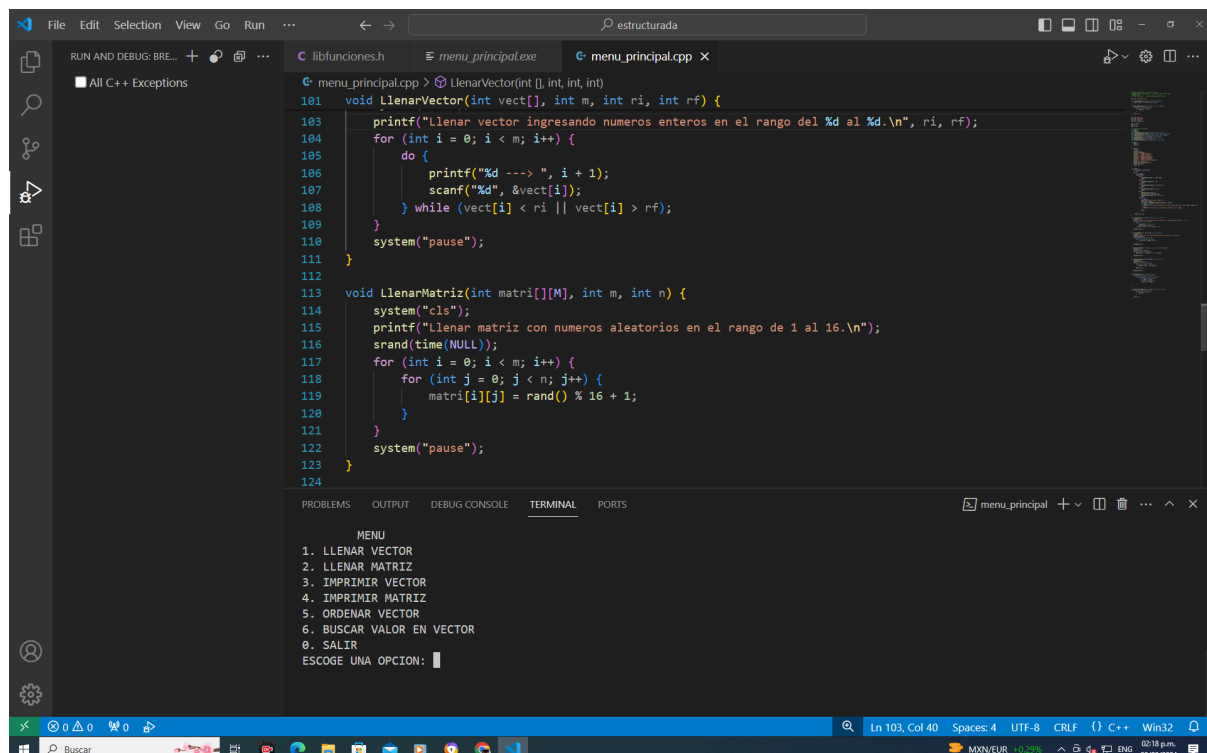
5. ORDENAR VECTOR

6. BUSCAR VALOR EN VECTOR

0. SALIR

ESCOGE UNA OPCION: 1

-2.- LLENAR MATRIZ .- Llenar la matriz de 4x4 con con números generados aleatoriamente, números entre el rango de 1 al 16 (**no repetidos**)



```
1  menu_principal.cpp > LlenarVector(int [], int, int, int)
2
3  void LlenarVector(int vect[], int m, int ri, int rf) {
4      printf("Llenar vector ingresando numeros enteros en el rango del %d al %d.\n", ri, rf);
5      for (int i = 0; i < m; i++) {
6          do {
7              printf("%d ---> ", i + 1);
8              scanf("%d", &vect[i]);
9              while (vect[i] < ri || vect[i] > rf);
10         } while (true);
11     }
12     system("pause");
13 }
14
15 void LlenarMatriz(int matriz[][M], int m, int n) {
16     system("cls");
17     printf("Llenar matriz con numeros aleatorios en el rango de 1 al 16.\n");
18     srand(time(NULL));
19     for (int i = 0; i < m; i++) {
20         for (int j = 0; j < n; j++) {
21             matriz[i][j] = rand() % 16 + 1;
22         }
23     }
24     system("pause");
25 }
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

menu_principal

MENU

1. LLENAR VECTOR

2. LLENAR MATRIZ

3. IMPRIMIR VECTOR

4. IMPRIMIR MATRIZ

5. ORDENAR VECTOR

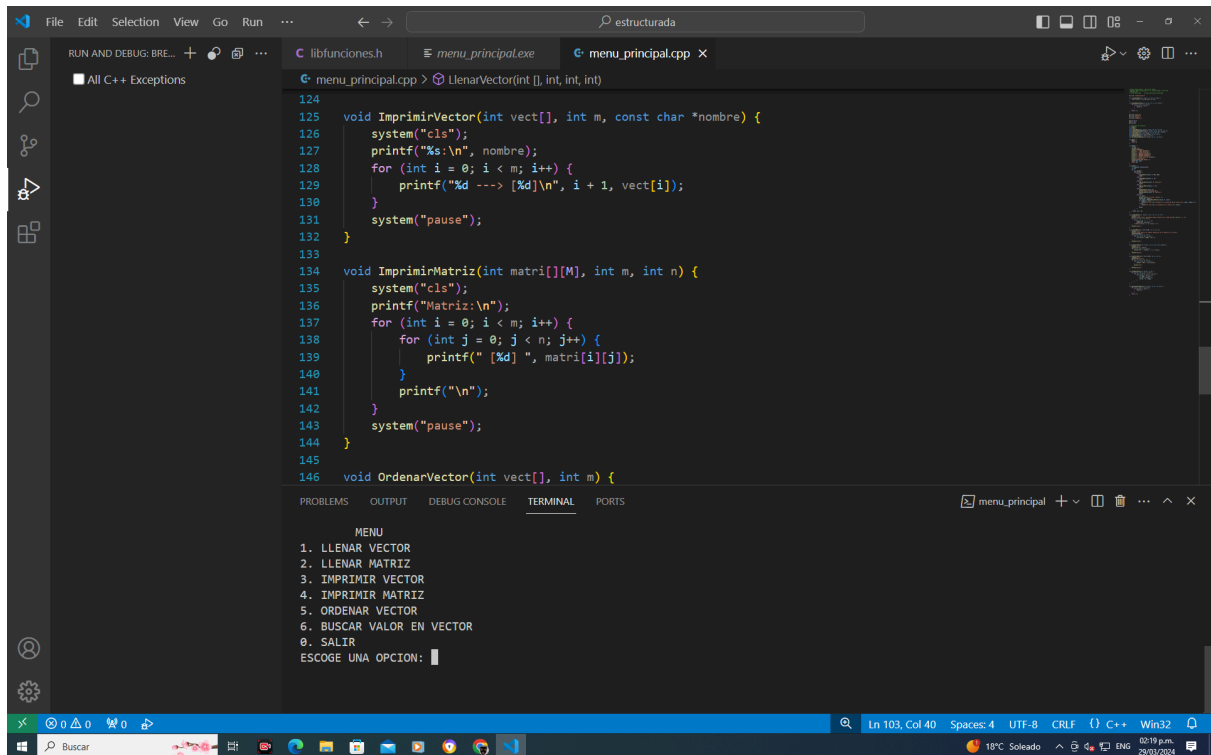
6. BUSCAR VALOR EN VECTOR

0. SALIR

ESCOGE UNA OPCION: 2

3.- IMPRIMIR VECTOR .- Imprime el vector que se envíe, donde la función recibe como parámetro el vector,tamaño, nombre del vector.

4.- IMPRIMIR MATRIZ.- Imprime la matriz sin importar el tamaño de la matriz recibiendo como parámetros la matriz, la cantidad de renglones y columnas, así como nombre que se le dará a la matriz



```
124
125 void ImprimirVector(int vect[], int m, const char *nombre) {
126     system("cls");
127     printf("%s:\n", nombre);
128     for (int i = 0; i < m; i++) {
129         printf("%d ---> [%d]\n", i + 1, vect[i]);
130     }
131     system("pause");
132 }
133
134 void ImprimirMatriz(int matri[][M], int m, int n) {
135     system("cls");
136     printf("Matriz:\n");
137     for (int i = 0; i < m; i++) {
138         for (int j = 0; j < n; j++) {
139             printf(" [%d] ", matri[i][j]);
140         }
141         printf("\n");
142     }
143     system("pause");
144 }
145
146 void OrdenarVector(int vect[], int m) {
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

menu_principal

```

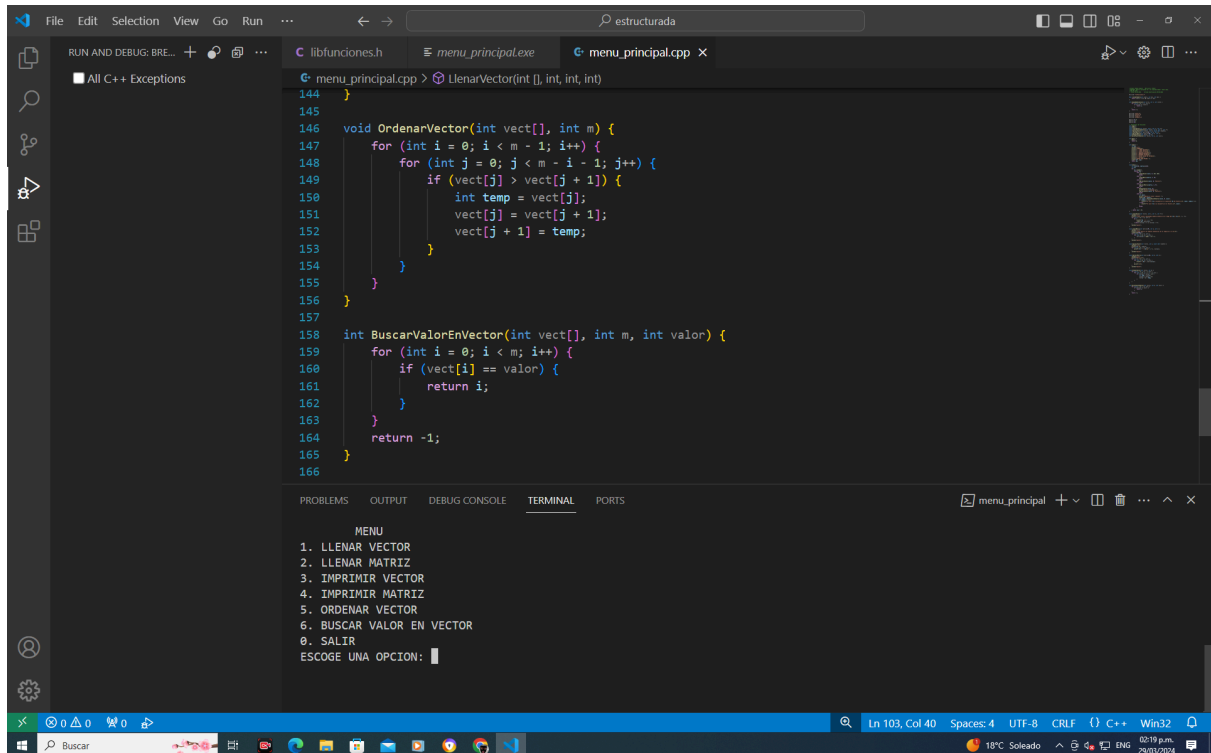
MENU
1. LLENAR VECTOR
2. LLENAR MATRIZ
3. IMPRIMIR VECTOR
4. IMPRIMIR MATRIZ
5. ORDENAR VECTOR
6. BUSCAR VALOR EN VECTOR
0. SALIR
ESCOGE UNA OPCION: 
```

Ln 103, Col 40 Spaces: 4 UTF-8 CRLF () C++ Win32

18°C Soleado 03:19 p.m. 29/03/2024

5.- ORDENAR VECTOR.- Usar función que ordene el vector por el método de ordenación de la Burbuja mejorada.

6.- BUSCAR VALOR EN VECTOR.- Buscar un valor en el vector usando el método de búsqueda secuencial.



The screenshot shows a C++ IDE with the following code in `menu_principal.cpp`:

```
144 }
145
146 void OrdenarVector(int vect[], int m) {
147     for (int i = 0; i < m - 1; i++) {
148         for (int j = 0; j < m - i - 1; j++) {
149             if (vect[j] > vect[j + 1]) {
150                 int temp = vect[j];
151                 vect[j] = vect[j + 1];
152                 vect[j + 1] = temp;
153             }
154         }
155     }
156 }
157
158 int BuscarValorEnVector(int vect[], int m, int valor) {
159     for (int i = 0; i < m; i++) {
160         if (vect[i] == valor) {
161             return i;
162         }
163     }
164     return -1;
165 }
166 }
```

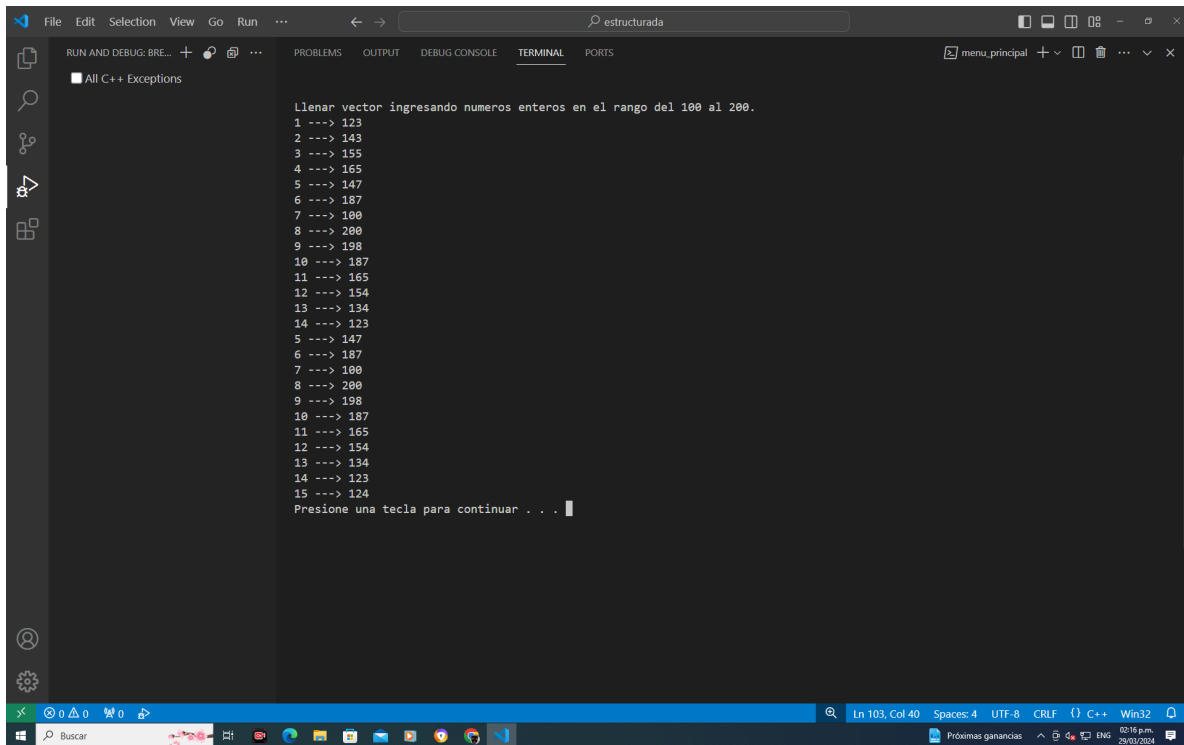
The terminal window shows the following menu:

```
MENU
1. LLENAR VECTOR
2. LLENAR MATRIZ
3. IMPRIMIR VECTOR
4. IMPRIMIR MATRIZ
5. ORDENAR VECTOR
6. BUSCAR VALOR EN VECTOR
0. SALIR
ESCOGE UNA OPCION: 
```

The status bar at the bottom indicates the cursor is at line 103, column 40, with 4 spaces, UTF-8 encoding, CRLF line endings, and C++ language. The system tray shows the date and time as 02/19 p.m. 29/03/2024.

EJECUCIÓN

-LLENAR VECTOR



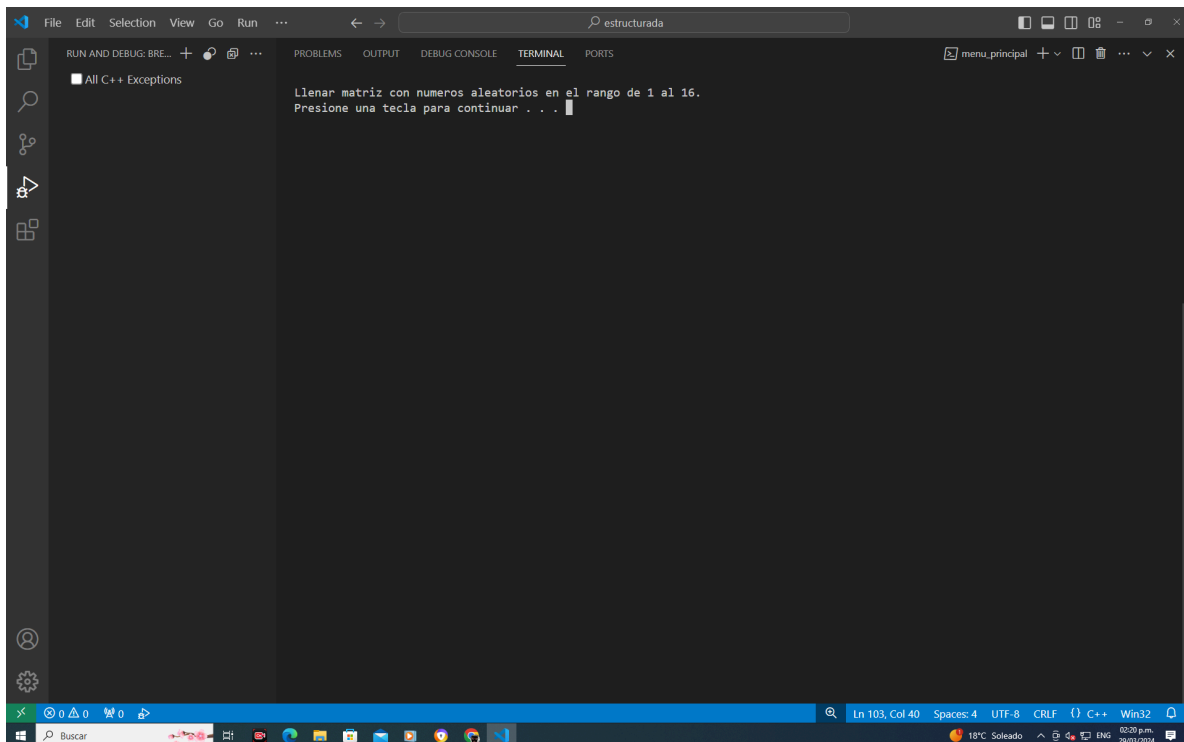
The screenshot shows a C++ IDE with a terminal window displaying the output of a program. The program prompts the user to fill a vector with integers in the range 100 to 200. The output shows 15 iterations, each displaying a random integer. The status bar at the bottom indicates the current line and column (Ln 103, Col 40) and the file encoding (UTF-8, CRLF).

```
File Edit Selection View Go Run ... estructura
RUN AND DEBUG: BRE... + ...
■ All C++ Exceptions

Llenar vector ingresando numeros enteros en el rango del 100 al 200.
1 ---> 123
2 ---> 143
3 ---> 155
4 ---> 165
5 ---> 147
6 ---> 187
7 ---> 100
8 ---> 200
9 ---> 198
10 ---> 187
11 ---> 165
12 ---> 154
13 ---> 134
14 ---> 123
5 ---> 147
6 ---> 187
7 ---> 100
8 ---> 200
9 ---> 198
10 ---> 187
11 ---> 165
12 ---> 154
13 ---> 134
14 ---> 123
15 ---> 124
Presione una tecla para continuar . . .
```

Ln 103, Col 40 Spaces: 4 UTF-8 CRLF {} C++ Win32

-LLENAR MATRIZ ALEATORIO



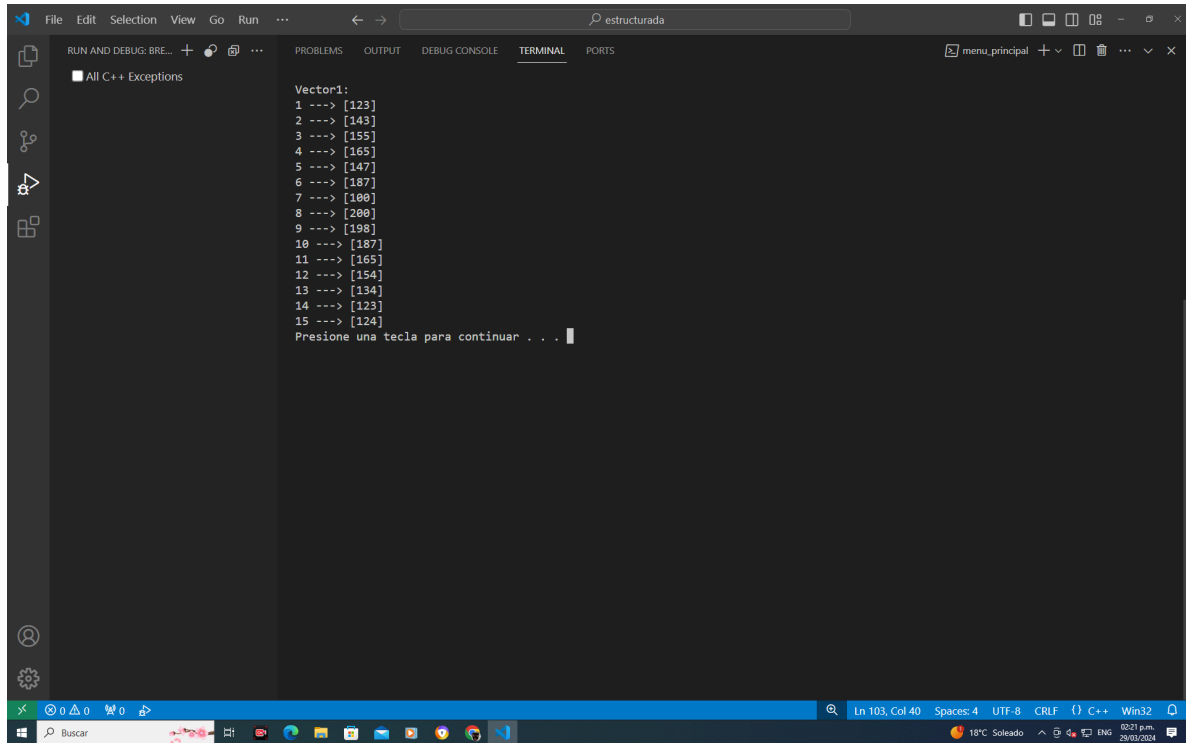
The screenshot shows a C++ IDE with a terminal window displaying the output of a program. The program prompts the user to fill a matrix with random numbers in the range 1 to 16. The output shows the prompt and the instruction to press a key to continue. The status bar at the bottom indicates the current line and column (Ln 103, Col 40) and the file encoding (UTF-8, CRLF).

```
File Edit Selection View Go Run ... estructura
RUN AND DEBUG: BRE... + ...
■ All C++ Exceptions

Llenar matriz con numeros aleatorios en el rango de 1 al 16.
Presione una tecla para continuar . . .
```

Ln 103, Col 40 Spaces: 4 UTF-8 CRLF {} C++ Win32

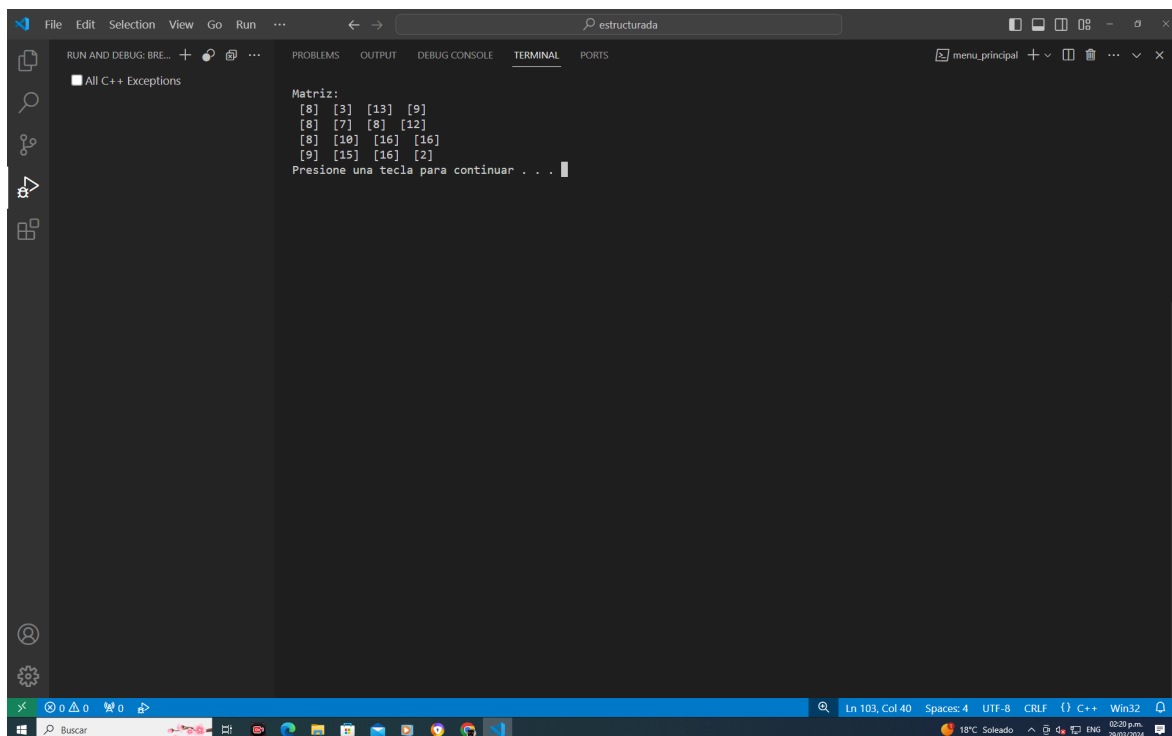
-IMPRIMIR VECTOR



The screenshot shows the Visual Studio Code interface with the 'TERMINAL' tab active. The terminal output displays a vector of 15 integers, each preceded by an index from 1 to 15. The values are: [123], [143], [155], [165], [147], [187], [100], [200], [198], [187], [165], [154], [134], [123], and [124]. The prompt 'Presione una tecla para continuar . . .' is visible at the bottom of the output.

```
Vector1:  
1 ----> [123]  
2 ----> [143]  
3 ----> [155]  
4 ----> [165]  
5 ----> [147]  
6 ----> [187]  
7 ----> [100]  
8 ----> [200]  
9 ----> [198]  
10 ----> [187]  
11 ----> [165]  
12 ----> [154]  
13 ----> [134]  
14 ----> [123]  
15 ----> [124]  
Presione una tecla para continuar . . .
```

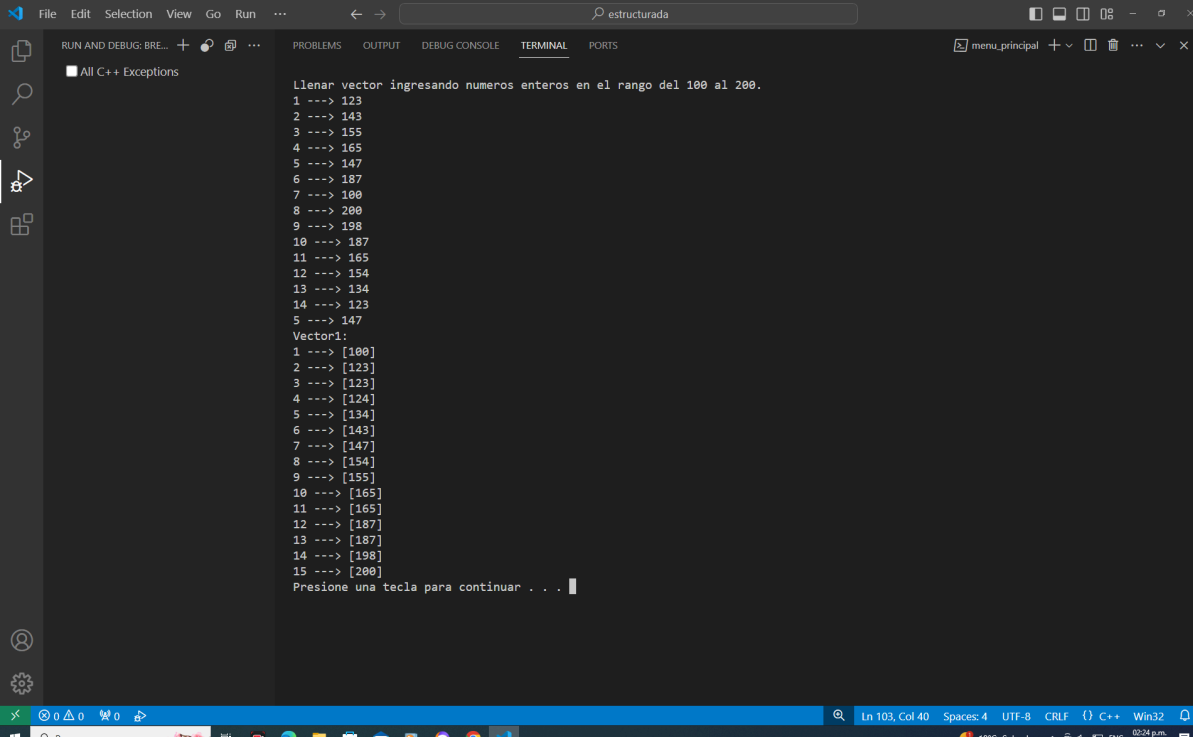
-IMPRIMIR MATRIZ



The screenshot shows the Visual Studio Code interface with the 'TERMINAL' tab active. The terminal output displays a 4x4 matrix of integers. The values are: [8], [3], [13], [9] in the first row; [8], [7], [8], [12] in the second row; [8], [10], [16], [16] in the third row; and [9], [15], [16], [2] in the fourth row. The prompt 'Presione una tecla para continuar . . .' is visible at the bottom of the output.

```
Matriz:  
[8] [3] [13] [9]  
[8] [7] [8] [12]  
[8] [10] [16] [16]  
[9] [15] [16] [2]  
Presione una tecla para continuar . . .
```

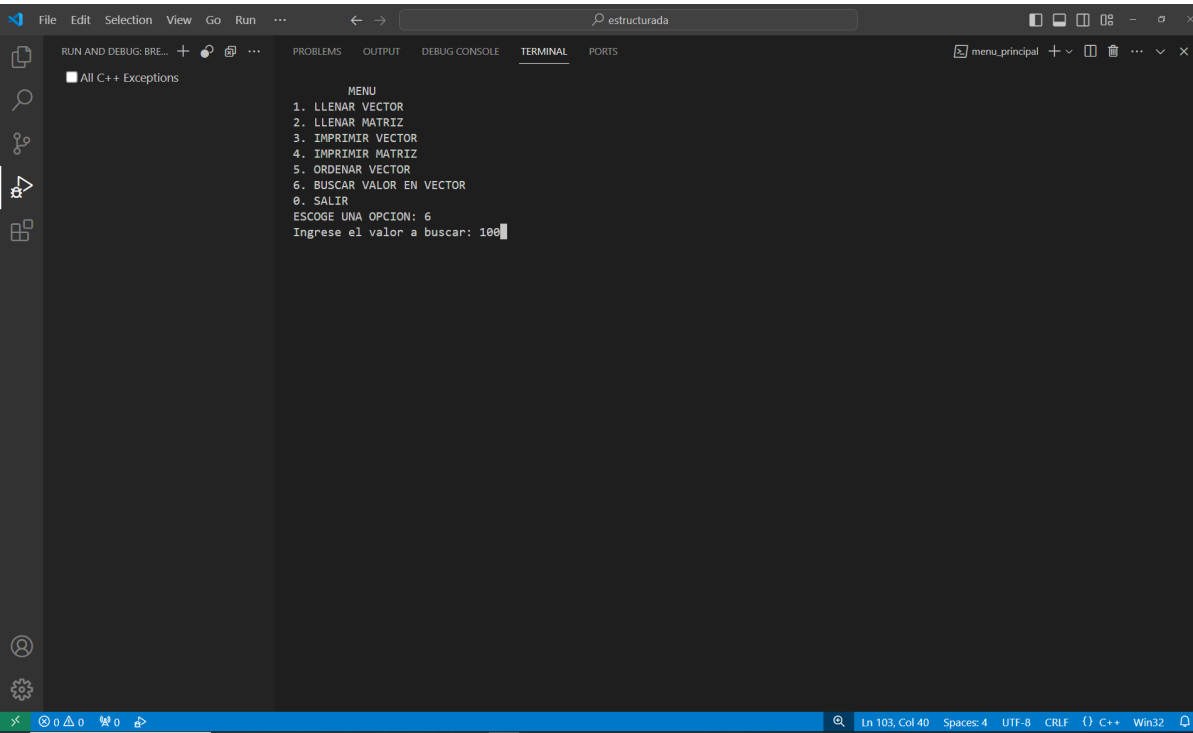
-ORDENAR VECTOR



The screenshot shows a C++ IDE with a terminal window. The terminal output displays the process of filling a vector with 15 integers and then sorting it. The numbers are: 123, 143, 155, 165, 147, 187, 100, 200, 198, 187, 165, 154, 134, 123, 147. After sorting, the vector contains: 100, 123, 123, 124, 134, 143, 147, 154, 155, 165, 165, 187, 187, 198, 200. The program prompts the user to press a key to continue.

```
llenar vector ingresando numeros enteros en el rango del 100 al 200.
1 ---> 123
2 ---> 143
3 ---> 155
4 ---> 165
5 ---> 147
6 ---> 187
7 ---> 100
8 ---> 200
9 ---> 198
10 ---> 187
11 ---> 165
12 ---> 154
13 ---> 134
14 ---> 123
15 ---> 147
Vector1:
1 ---> [100]
2 ---> [123]
3 ---> [123]
4 ---> [124]
5 ---> [134]
6 ---> [143]
7 ---> [147]
8 ---> [154]
9 ---> [155]
10 ---> [165]
11 ---> [165]
12 ---> [187]
13 ---> [187]
14 ---> [198]
15 ---> [200]
Presione una tecla para continuar . . .
```

-BUSCAR VECTOR



The screenshot shows a C++ IDE with a terminal window. The terminal output displays a menu with 6 options: 1. LLENAR VECTOR, 2. LLENAR MATRIZ, 3. IMPRIMIR VECTOR, 4. IMPRIMIR MATRIZ, 5. ORDENAR VECTOR, 6. BUSCAR VALOR EN VECTOR. The user has selected option 6, and the program prompts the user to enter a value to search for. The user has entered 100.

```
MENU
1. LLENAR VECTOR
2. LLENAR MATRIZ
3. IMPRIMIR VECTOR
4. IMPRIMIR MATRIZ
5. ORDENAR VECTOR
6. BUSCAR VALOR EN VECTOR
0. SALIR
ESCOGE UNA OPCION: 6
Ingrese el valor a buscar: 100
```

GIT HUB

https://github.com/sergiofabian05/SFA_PE_ACT9.git