



Universidad Autónoma de Baja California

Facultad de Ingeniería Arquitectura y Diseño

Ingeniero en computación

Ingeniero en Software y tecnologías emergentes

Materia: Programación Estructurada / Clave 36276

Alumno: Kevin Alejandro Gonzalez Torres

Matrícula: 372354

Maestro: Pedro Núñez Yépiz

Actividad Numero: 8

Tema - Unidad: Vectores y matrices

Ensenada Baja California a 1 de octubre del 2022



Universidad Autónoma de Baja California

Facultad de Ingeniería Arquitectura y Diseño

1. INTRODUCCIÓN

En este reporte se mostrará el uso de vectores y matrices, de sus distintos usos en diversas aplicaciones diarias y de cómo funcionan.

2. COMPETENCIA

Se practicará y se aprenderá el uso de los vectores y las matrices.

3. FUNDAMENTOS

Vectores:

<https://info64.ro/en/Vectors/>

https://es.wikibooks.org/wiki/Programaci%C3%B3n_en_C/Vectores

Matrices:

<https://beginnersbook.com/2014/01/2d-arrays-in-c-example/>

<https://www.programiz.com/c-programming/c-multi-dimensional-arrays>

4. PROCEDIMIENTO

Realiza programa en C el programa deberá tener el siguiente menú.

MENÚ

- 1.- LLENAR VECTOR
- 2.- LLENAR MATRIZ
- 3.- IMPRIMIR VECTOR
- 4.- IMPRIMIR MATRIZ
- 5.- ORDENAR VECTOR
- 6.- BUSCAR VALOR EN VECTOR
- 0.- SALIR

NOTA: El programa deberá repetirse cuantas veces lo desee el usuario, Validado el menú con la función vali_num



5. RESULTADOS Y CONCLUSIONES

1.-

```
28 void fillVectorRandom(int vector[], int size, int ri, int rf)
29 {
30     int checkRepeated;
31
32     srand(time(NULL));
33     for (int i = 0; i < size; i++)
34     {
35         // function to avoid repeated number
36         do
37         {
38             checkRepeated = 0; // initialize to assume the number is not repeated
39             vector[i] = rand() % (rf - ri + 1) + ri;
40
41             // check if the random number already exists in the vector
42             for (int j = 0; j < i; j++)
43             {
44                 if (vector[j] == vector[i])
45                 {
46                     checkRepeated = 1; // set to 1 if the number is repeated
47                 }
48             }
49         } while (checkRepeated); // repeat until a unique number is generated
50     }
51 }
```

2.-

```
67 void fillMatrixRandom(int matrix[][5], int vector[], int r, int c)
68 {
69     int k = 0;
70     fillVectorRandom(vector, 16, 1, 16);
71     for (int i = 0; i < r; i++)
72     {
73         for (int j = 0; j < c; j++)
74         {
75             matrix[i][j] = vector[k++];
76         }
77     }
78 }
```



Universidad Autónoma de Baja California

Facultad de Ingeniería Arquitectura y Diseño

3.-

```
53 void printVector(int vector[], int size)
54 {
55     printf("Vector elements: {");
56     for (int i = 0; i < size; i++)
57     {
58         printf("%d", vector[i]);
59         if (i < size - 1) // separates each number with a comma
60         {
61             printf(", ");
62         }
63     }
64     printf("}\n");
65 }
```

4.-

```
80 void printMatrix(int matrix[][5], int size) // doesnt avoid repeated numbers
81 {
82     int r, c;
83
84     printf("Matrix elements: \n{\n");
85     for (r = 0; r < size; r++)
86     {
87         for (c = 0; c < size; c++)
88         {
89             printf("%d\t", matrix[r][c]);
90         }
91         printf("\n");
92     }
93     printf("}");
94 }
```



Universidad Autónoma de Baja California

Facultad de Ingeniería Arquitectura y Diseño

5.-

```
96 void sortVector(int vector[], int size)
97 {
98     int temp;
99
100     for (int i = 0; i < size - 1; i++)
101     {
102         for (int j = i + 1; j < size; j++)
103         {
104             if (vector[j] < vector[i])
105             {
106                 temp = vector[i];
107                 vector[i] = vector[j];
108                 vector[j] = temp;
109             }
110         }
111     }
112 }
```

6.-

```
114 int linearSearch(int vector[], int size, int searchNumber)
115 {
116     int i;
117
118     for (i = 0; i < size; i++)
119     {
120         if (vector[i] == searchNumber)
121         {
122             return i;
123         }
124     }
125     return -1;
126 }
127
```



Universidad Autónoma de Baja California

Facultad de Ingeniería Arquitectura y Diseño

6. ANEXOS

GTKA_RP09_PE_ANEXOS



Universidad Autónoma de Baja California

Facultad de Ingeniería Arquitectura y Diseño

7. REFERENCIAS

Diseño de algoritmos y su codificación en lenguaje C

Corona, M.A. y Ancona, M.A. (2011)..

España: McGraw-Hill.

ISBN: 9786071505712

Programación estructurada a fondo: implementación de algoritmos en C

:Pearson Educación. Sznajdleder, P. A. (2017)..

Buenos Aires, Argentina: Alfaomega

Como programar en C/C++

H.M. Deitel/ P.J. Deitel

Segunda edición

Editorial: Prentice Hall.

ISBN: 9688804711

Programación en C. Metodología, estructura de datos y objetos

Joyanes, L. y Zahonero, I. (2001)..

España: McGraw-Hill.

ISBN: 8448130138