



Universidad Autónoma de Baja California

Facultad de Ingeniería Arquitectura y Diseño

Ingeniero en Software y tecnologías emergentes

Materia: Programación Estructurada / Clave 36276

Alumno: Antonio Ramos González

Matrícula: 372576

Maestro: Pedro Núñez Yépiz

Actividad No. 8 : Arreglos en c

Tema - Unidad 5 : arreglos y cadenas

Ensenada Baja California a 1 de octubre del 2023



Universidad Autónoma de Baja California

Facultad de Ingeniería Arquitectura y Diseño

1. INTRODUCCIÓN

Los arreglos son una estructura fundamental que permite el almacenamiento de distintas colecciones de datos del mismo tipo bajo un mismo nombre, la almacenación de dichos datos dentro de un vector facilita enormemente su acceso y manipulación, por lo que son sumamente útiles cuando se trabaja con una gran cantidad de datos

2. COMPETENCIA

Desarrollar conocimientos sobre la construcción y visualización tanto de vectores, como de matrices. Siguiendo reforzando nuestros conocimientos previos como la validación, el uso de distintos ciclos y estructuras de control.



Universidad Autónoma de Baja California

Facultad de Ingeniería Arquitectura y Diseño

3. FUNDAMENTOS

Los arreglos son colecciones de variables del mismo tipo donde se guarda todo tipo de información, a la cual se accede mediante un índice, el cual indica en donde se guarda dicha información, siendo la dirección más baja el primer elemento y la más alta el último elemento.

Existen diferentes dimensiones que puede tener un arreglo, siendo el arreglo unidimensional el más básico de todos, el cual se asigna de la siguiente manera

Int vector[tamaño]

Los elementos de los arreglos en direcciones controladas por el tamaño, dichas direcciones iniciaran en 0 y terminaran en n-1, por ejemplo

Int vect[10](ira del 0 al 9)

La asignación de un valor al arreglo se presenta de la siguiente forma

Int vect[10]

I=5

Vect [i]= 255 (i representando la dirección del índice en donde se guardará la información)

Los arreglos bidimensionales, también conocidos como matrices son un tipo de arreglo que almacena información guardada en diferentes filas y columnas.

Int matriz [filas][columnas]

La manera en que las matrices leen sus datos es de izquierda a derecha, leyendo primero las fila y posteriormente las columnas, cuando llega a la última columna regresa a la columna 1 pero ahora se encuentra en la siguiente fila

Int mtz[4][4]

I=65

Mtz[1][3]=i(asignando el valor de i en la fila 1 columna 3)

Nuñez,Yepiz,P.(2022)Programación estructurada



Universidad Autónoma de Baja California

Facultad de Ingeniería Arquitectura y Diseño

4. PROCEDIMIENTO

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

MENÚ

- 1.- LLENAR **VECTOR 1** (MANUALMENTE)
- 2.- LLENAR **VECTOR 2** ALEATORIAMENTE
- 3.- LLENAR **VECTOR 3** (CON VECTOR1 Y VECTOR2)
- 4.- IMPRIMIR VECTORES
- 5.- LLENA MATRIZ 4 X 4
- 6.- IMPRIMIR MATRIZ
- 0.- SALIR

NOTA: EL PROGRAMA DEBERÁ REPETIRSE CUANTAS VECES LO DESEE EL USUARIO

NOTA 2: EL **VECTOR 1** DE 10 POSICIONES, NÚMEROS DEL **30 AL 70**

NOTA 3: EL **VECTOR 2** DE 10 POSICIONES CON NÚMEROS GENERADOS ALEATORIAMENTE DEL 1 AL 20 (SIN REPETIR)

NOTA 4: EL **VECTOR 3** DE 20 POSICIONES, CON LOS DATOS DEL ARREGLO1 Y ARREGLO2

NOTA 5: MATRIZ 4 X 4 LLENARLA CON LOS DATOS DEL VECTOR1 Y VECTOR2,

AQUÍ SE PONE LA PRÁCTICA QUE SE REALIZARÁ, ENUNCIADO O PLANTEAMIENTO DEL MAESTRO PARA LA PRÁCTICA



5. RESULTADOS Y CONCLUSIONES

En conclusión, los vectores y las matrices son estructuras clave para la organización de los distintos tipos de datos, volviendo más eficiente la manera de trabajar cuando se tiene una gran cantidad de datos, permitiéndonos resolver una gran cantidad de problemas, siendo una herramienta esencial para nosotros los programadores

```
1 // Antonio Ramos Gonzalez Mt: 372576
2 // 9/27/2023 || 10/01/2023
3 // Funciones e Introducción a Arreglos en C
4 // RGA_Act8_932
5 #include <stdio.h>
6 #include <time.h>
7 #include <stdlib.h>
8 #include <string.h>
9
10 int msg();
11 void menu();
12 int validar(const char msg[], int li_in, int li_sup);
13 void vect_1(int vect1[], int m);
14 void vect_2(int vect2[], int lim_in, int lim_sup);
15 void vect_3(int vect1[], int vect2[], int vect3[], int m);
16 void imprimir_vectores(int vect1[], int vect2[], int vect3[], int m, int n);
17 void matriz(int vect1[], int vect2[], int mtz[4][4], int m, int v);
18 void imprimir_matriz(int mtz[4][4], int m);
19
20 > int main() ...
26
27 > int msg() ...
43
44 void menu()
45 {
46     int vect1[10], vect2[10], vect3[20]; // definimos el tamaño de los vectores
47     int mtz[4][4]; // definimos el tamaño de la matriz
48     short int opc;
49     do
50     {
51         opc = msg(); // leemos el valor retornado de la función msg
52         switch (opc)
53         {
54             case 1:
55                 vect_1(vect1, 10); // se manda vect1, así como su tamaño a la función
56
57                 break;
58             case 2:
59                 vect_2(vect2, 1, 20); // se manda vect2, así como su límite de datos a la función
60                 break;
61             case 3:
62                 vect_3(vect1, vect2, vect3, 20); // se manda vect1, vect2 para llenar vect3
63                 break;
64             case 4:
65                 imprimir_vectores(vect1, vect2, vect3, 10, 20); // se manda vect1, vect2 y vect3 para imprimirlos
66                 break;
67             case 5:
68                 matriz(vect1, vect2, mtz, 4, 10); // se manda vect1, vect2 para llenar mtz
69                 break;
70             case 6:
71                 imprimir_matriz(mtz, 4); // se manda mtz para imprimirla
72                 break;
73             default:
74                 break;
75         }
76     } while (opc != 0);
77 }
78
79
```



Universidad Autónoma de Baja California

Facultad de Ingeniería Arquitectura y Diseño

6. 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