



Ingeniero en Software y tecnologías emergentes

Materia: Programación Estructurada / Clave 36276

Alumno: Gabriel Tejeda Chávez

Matrícula: 347723

Maestro: Pedro Núñez Yépiz

Actividad No 8

Tema – Arreglos y funciones

Ensenada Baja California a 24 de marzo del 2024



Universidad Autónoma de Baja California

Facultad de Ingeniería Arquitectura y Diseño

Introducción

Los arreglos son estructuras fundamentales en la programación, permitiendo almacenar y acceder a múltiples elementos de datos bajo un mismo nombre. Representan una colección ordenada de elementos del mismo tipo, lo que los convierte en herramientas poderosas para organizar información de manera eficiente en la memoria de una computadora.

La eficiencia en el acceso a los elementos es una de las características más destacadas de los arreglos. Al poder acceder directamente a un elemento a través de su posición en el arreglo, los algoritmos pueden ejecutarse de manera más rápida y con menor consumo de recursos en comparación con otras estructuras de datos. Esta capacidad de acceso aleatorio facilita la implementación de algoritmos de búsqueda, ordenación y manipulación de datos de manera eficiente.

En el siguiente reporte se evidenciarán diversos casos donde se resuelven problemas mediante la aplicación de arreglos.

Competencias

De esta practica se espera aprender el como trabajar con arreglos y comprender las estructuras de datos básicas para el diseño e implementación de algoritmos eficientes, además de que por medio de la ejecución podamos desarrollar habilidades analíticas y de resolución de problemas.

Fundamentos

Los arreglos (arrays) son variables del mismo tipo de dato que tienen el mismo nombre y que se distinguen y referencian por un índice.

Sintaxis:

`<tipo> <variable> [N]`

Se declara un arreglo de nombre `<variable>` con `N` elementos de tipo `<tipo>`, (`N` es una constante).

Ejemplo: `int a[10];`

Los arreglos (arrays) permiten almacenar vectores y matrices. Los arreglos unidimensionales sirven para manejar vectores y los arreglos bidimensionales para matrices. Sin embargo, las matrices también se pueden almacenar mediante arreglos unidimensionales y por medio de apuntadores a apuntadores.



Universidad Autónoma de Baja California

Facultad de Ingeniería Arquitectura y Diseño

La palabra unidimensional no indica que se trata de vectores en espacios de dimensión uno; indica que su manejo se hace mediante un subíndice. El manejo de los arreglos bidimensionales se hace mediante dos subíndices.

Cuando un parámetro de una función es un arreglo, se considera implícitamente que es un parámetro por referencia. O sea, si en la función se modifica algún elemento del arreglo, entonces se modificó realmente el valor original y no una copia.

Cuando en una función un parámetro es un arreglo bidimensional, la función debe saber, en su definición, el número de columnas del arreglo bidimensional. Por eso en la definición de las funciones está a `[[40]`.

Esto hace que las funciones del ejemplo sirvan únicamente para arreglos bidimensionales definidos con 40 columnas. Entonces estas funciones no son de uso general. Este inconveniente se puede resolver de dos maneras:

Mediante apuntadores y apuntadores dobles.

Almacenando las matrices en arreglos unidimensionales con la convención de que los primeros elementos del arreglo corresponden a la primera fila de la matriz, los que siguen corresponden a la segunda fila, y así sucesivamente. Esta modalidad es muy usada, tiene algunas ventajas muy importantes.

En resumen, los arreglos bidimensionales no son muy adecuados para pasarlos como parámetros a funciones. Su uso debería restringirse a casos en que el arreglo bidimensional se usa únicamente en la función donde se define.



Universidad Autónoma de Baja California

Facultad de Ingeniería Arquitectura y Diseño

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.



Resultados y Conclusiones

```
84 void llenar_vector(int vect1[], int m)
85 {
86     system("CLS");
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 }
```

Esta función básicamente lo que hace es recibir un vector de n tamaño y preguntarle al usuario el numero que desea ingresar al vector en la posición i, ya que al ser un arreglo hay que especificar el índice donde se quiere guardar el valor y el método más efectivo es iterando el arreglo. Al haber una restricción de números utilice un "if" para que detectara cuando el usuario ingresara un numero fuera del rango, al hacer esto se imprime un mensaje de error y se vuelve a ejecutar el arreglo en la misma posición hasta que ingrese el numero correcto.



Universidad Autónoma de Baja California

Facultad de Ingeniería Arquitectura y Diseño

```
void vector_num_aleat(int vect2[], int m)
{
    system("CLS");
    int i, j, k, repetido, num;
    srand(time(NULL));

    i = 0;
    while (i < m)
    {
        num = (rand() % 20) + 1;
        if (buscador(vect2, num, m) == 0)
        {
            vect2[i] = num;
            i++;
        }
    }
}
```

Para generar los números aleatorios utilice la función rand de la librería <time.h>, también cree una función que recibe al arreglo y busca en el si algún numero se repite.

```
int buscador(int vect2[], int num, int m)
{
    int i;
    for (i = 0; i < m; i++)
    {
        if (vect2[i] == num)
        {
            return 1;
        }
    }
    return 0;
}
```

Y en caso de ser verdad retorna 1. Si retorna uno se genera un nuevo numero aleatorio hasta que ya no encuentre el mismo número, solo en ese caso la iteración aumentara en uno la posición del índice.



Universidad Autónoma de Baja California

Facultad de Ingeniería Arquitectura y Diseño

```
void Llenar_matriz(int vect1[], int vect2[], int m)
{
    system("CLS");
    int matriz[2][N];
    int i, j;

    for (i = 0; i < m; i++)
    {
        matriz[0][i] = vect1[i];
    }

    for (i = 0; i < m; i++)
    {
        matriz[1][i] = vect2[i];
    }
}
```

Para llenar la matriz utilice dos ciclos "for", cuidando llenar una fila con un vector y la otra con el segundo vector.

Conclusión

Trabajar con arreglos ha ampliado mi comprensión de las estructuras de datos básicas y ha fortalecido mis habilidades en resolución de problemas, pensamiento algorítmico y optimización de código.

Creo que su comprensión es fundamental para futuros temas por lo que, lo visto en esta práctica será de utilidad para futuros proyectos.



Universidad Autónoma de Baja California

Facultad de Ingeniería Arquitectura y Diseño

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