



Carátula para entrega de prácticas

Facultad de Ingeniería

Laboratorio de docencia

Laboratorios de computación salas A y B

Profesor:

Ingeniera Claudia Rodriguez Espino

Asignatura:

Fundamentos de Programación

Grupo:

1102

No de Práctica(s):

Práctica número 11

Integrante(s):

Chaveste Bermejo Carlos Alberto

Semestre:

2018-1

Fecha de entrega:

03/11/2017

Observaciones:

CALIFICACIÓN: _____

OBJETIVO:

Reconocer la importancia y utilidad de los arreglos, en la elaboración de programas que resuelvan problemas que requieran agrupar datos del mismo tipo, así como trabajar con arreglos tanto unidimensionales como multidimensionales.

DESARROLLO:

En la práctica, aprendimos sobre los arreglos que son un conjunto de datos, variables, números o caracteres agrupados en un conjunto con un tamaño determinado por el programador. Cuando se inicializa un arreglo se hace de la manera siguiente:

Tipo de dato nom_arreglo[Tamaño del arreglo];

Cada uno de los elementos del conjunto tiene una dirección asociada, por lo que para poder hablar de elementos en los arreglos, se debe de especificar la posición en la que se encuentra el elemento. Esto se expresa de la manera:

nom_arreglo[posición del elemento];

De la misma forma, los arreglos pueden ser de distintas dimensiones, esto sirve como una matriz, en la que si se tiene una dimensión, solo se trata de una columna, pero cuando son dos dimensiones, se expresa en varias filas y columnas. Cabe aclarar que si se hace un arreglo de 3 o más dimensiones, entonces solo se hacen un tipo de “sub-arreglos” dentro de otros arreglos. Su sintaxis es:

tipoDato nombre[tamaño][tamaño]...[tamaño];

ACTIVIDAD EN CLASE

Programa 1.Matriz por escalar.

```
#include<stdio.h>
#include<conio.h>
int i=0;
int j=0;
int x=0;
int y=0;
int esc;

main()
{
printf("\t\t\tPrograma para multiplicar una matriz por una escalar\n\n");

printf("Dame el valor de la escalar\n");
scanf("%d",&esc);

printf("Cuantas columnas quieres?");
scanf("%d",&x);

printf("Cuantas filas quieres?");
scanf("%d",&y);

int matriz[x][y];
for(i=0;i<x;i++){
for(j=0;j<y;j++){
printf("Dame el valor para cada casilla [%d][%d]",i,j);
scanf("%d",&matriz[i][j]);
}
}
printf("\nOriginal");
for(i=0;i<x;i++){
printf("\t\n");
for(j=0;j<y;j++){
printf("%d\t",matriz[i][j]);
}
}
printf("\t\n");
int resultante[i][j];
```

```

printf("\nMultiplicacion de la matriz por la escalar:");
for(i=0;i<x;i++){
printf("\t\n");
for(j=0;j<y;j++){
resultante[x][y] = matriz[i][j]*esc;
printf("%d\t",resultante[x][y]);
}
}

printf("\n\t");
getch();
}

```

```

C:\Users\SATELLITE\Desktop\Programas Fundamentos\programas ordenados\Prog 22 Matriz por Esc...
Programa para multiplicar una matriz por una escalar

Dame el valor de la escalar
2
Cuántas columnas quieres?2
Cuántas filas quieres?2
Dame el valor para cada casilla [0][0]2
Dame el valor para cada casilla [0][1]2
Dame el valor para cada casilla [1][0]2
Dame el valor para cada casilla [1][1]2

Original
2      2
2      2

Multiplicacion de la matriz por la escalar:
4      4
4      4

-----
Process exited after 7.877 seconds with return value 0
Presione una tecla para continuar . . .

```

Programa 2. Suma de matrices

```

#include<stdio.h>
main()
{
int mA[15],mB[5],mC[15],cont;

for(cont=0;cont<5;cont++)
{
printf("introduce 5 valores para la primer matriz\n");
scanf("%d",&mA[cont]);
}

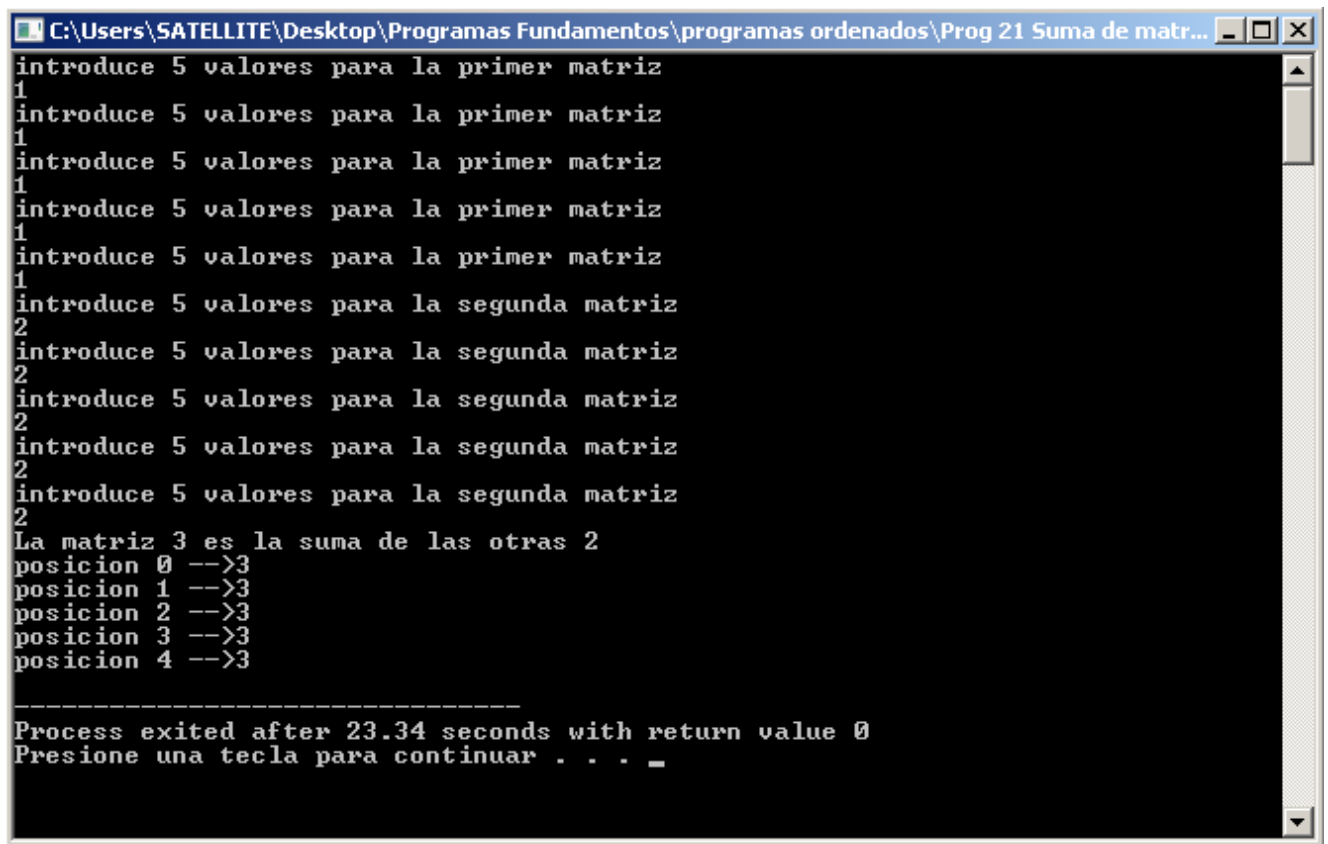
```

```

for(cont=0;cont<5;cont++)
    {
        printf("introduce 5 valores para la segunda matriz\n");
        scanf("%d",&mB[cont]);
    }

    printf("La matriz 3 es la suma de las otras 2\n");
for(cont=0;cont<5;cont++)
    {
        mC[cont]=mA[cont]+mB[cont];
        printf("posicion %d -->%d\n",cont,mC[cont]);
    }
}

```



```

C:\Users\SATELLITE\Desktop\Programas Fundamentos\programas ordenados\Prog 21 Suma de matr...
introduce 5 valores para la primer matriz
1
introduce 5 valores para la primer matriz
1
introduce 5 valores para la primer matriz
1
introduce 5 valores para la primer matriz
1
introduce 5 valores para la primer matriz
1
introduce 5 valores para la segunda matriz
2
introduce 5 valores para la segunda matriz
2
introduce 5 valores para la segunda matriz
2
introduce 5 valores para la segunda matriz
2
introduce 5 valores para la segunda matriz
2
La matriz 3 es la suma de las otras 2
posicion 0 -->3
posicion 1 -->3
posicion 2 -->3
posicion 3 -->3
posicion 4 -->3

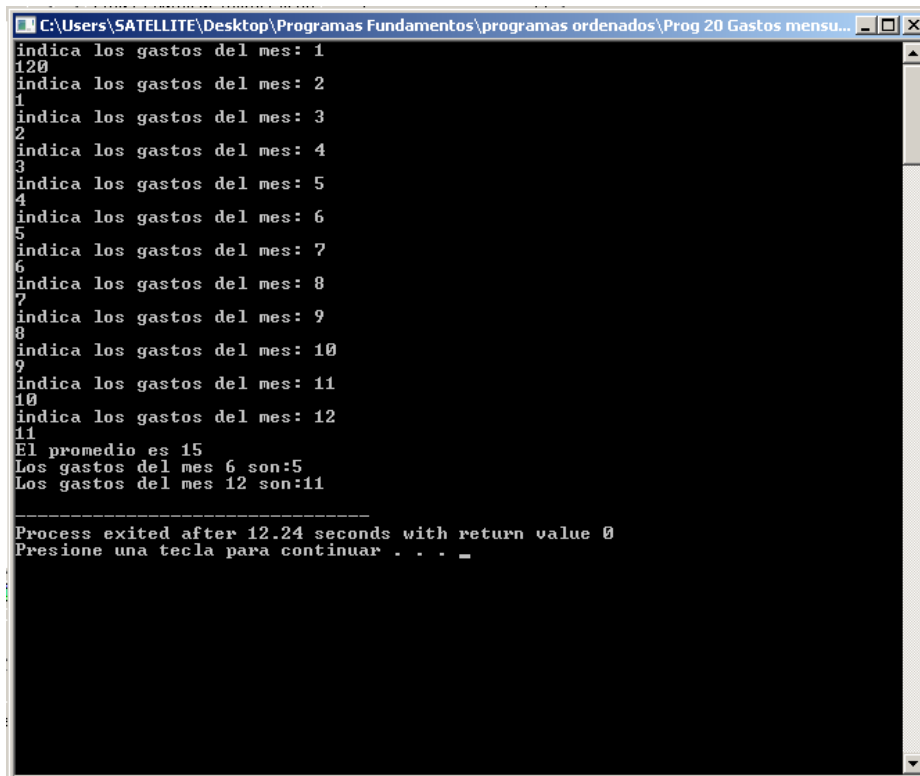
-----
Process exited after 23.34 seconds with return value 0
Presione una tecla para continuar . . . _

```

Programa 3. Gastos mensuales.

```
#include<stdio.h>
int mes[15],a,promedio;
main()
{

    for (a=1;a<13;a++)
    {
        printf("indica los gastos del mes: %d\n",a);
        scanf ("%d",&mes[a]);
    }
    for(a=1;a<13;a++)
    {
        promedio=promedio+mes[a];
    }
    promedio=promedio/12;
    printf("El promedio es %d\n",promedio);
    printf("Los gastos del mes 6 son:%d\n",mes[6]);
    printf("Los gastos del mes 12 son:%d\n",mes[12]);
}
```



```
C:\Users\SATELLITE\Desktop\Programas Fundamentos\programas ordenados\Prog 20 Gastos mensuales
indica los gastos del mes: 1
120
indica los gastos del mes: 2
1
indica los gastos del mes: 3
2
indica los gastos del mes: 4
3
indica los gastos del mes: 5
4
indica los gastos del mes: 6
5
indica los gastos del mes: 7
6
indica los gastos del mes: 8
7
indica los gastos del mes: 9
8
indica los gastos del mes: 10
9
indica los gastos del mes: 11
10
indica los gastos del mes: 12
11
El promedio es 15
Los gastos del mes 6 son:5
Los gastos del mes 12 son:11

-----
Process exited after 12.24 seconds with return value 0
Presione una tecla para continuar . . . _
```

CONCLUSIÓN

Los arreglos sirven para agrupar elementos, por lo que son muy importantes para distintas cosas, funciones y motivos, ya que estos son todos del mismo tipo, se puede usar si se quiere aumentar la eficiencia del programa; al igual que el tiempo en el que se crea un código.

BIBLIOGRAFÍA

<http://lcp02.fi-b.unam.mx/#>