

Práctica 3 – Lenguaje C

Contenido

- Arreglos
- Cadenas
- Estructuras bidimensionales
- Struct - Arreglo de structs

Práctico

1. El siguiente programa lee un arreglo y lo modifica eliminando el elemento de la posición central. Se sabe que se han cometido algunos errores en los parámetros y punteros (direcciones de). Corregirlo para que funcione como se espera

```
#include <stdio.h>
void leevector(int a[], int *dn);
void elimina_central(int a, int n);
void escvector(int n[], int a);

int main(void) {
    /* programa principal */
    int n;
    int a[20];
    leevector(&a, &n);
    elimina_central(a, *n);
    escvector(a[20], &n);
    return 0;
}

void leevector(int a[], int *dn) {
    int i;
    printf("Ingrese la cantidad de elementos del vector<=100: ");
    scanf("%d", *dn);
    for(i=0; i<*dn; i++) {
        printf("Ingrese el elemento %d: ", i+1);
        scanf("%d", a[i]);
    }
}

void elimina_central (int a, int n) {
    int i;
    for(i = (n + 1) / 2; i < n-1; i++)
        a[i] = a[i+1];
    n--;
}

void escvector(int n[], int a) {
    int i = 0;
    for( ; i < a; i++)
        printf("%d ", n[i]);
}
```

2. Indicar cuál es el efecto de las siguientes declaraciones:

a) int a[20]	b) int a[20], b[5]
c) int n[5] = {10, 20, 30, 40, 50}	d) float m[] = {4.7, 3.9, 2.0, 1.2, 0.6}
e) char c[4] = {'h', 'o', 'l', 'a'}	f) int d[6] = {1, 2, 3}
g) float *pf, f[10]; pf = f;	h) int a[5], b[5] = {0}

3. A partir de la declaración b) del ejercicio anterior, explicar si las siguientes sentencias son válidas. Justificar.

```
a = b;
a[5] = b[5];
a = {1, 2, 3, 4, 5};
a[5] = {1, 2, 3, 4, 5};
```

4. Indicar si el siguiente código es correcto o no, justificando debidamente.

```
char a[] = "hola";
char b[] = "mundo", c[10];
if(a==b)
    printf("Son iguales.");
else
    c = b;
```

5. ¿Cuál es la diferencia entre estas dos declaraciones?

```
char a[5];          char *a;
```

6. ¿Cuál es la diferencia entre las siguientes declaraciones?

```
char a[] = "hola";
char a[4] = "hola";
char a[] = {'h', 'o', 'l', 'a'};
char a[] = {'h', 'o', 'l', 'a', '\0'};
char *a = "hola";
```

7. Para cada una de las siguientes funciones describir los parámetros, qué hacen y qué archivo .h debe ser incluido: strlen, strcmp, strcpy, strcat

8. ¿Qué emite el siguiente programa?

```
#include <stdio.h>
#include <string.h>
int main() {
    char nota [] = "Nos veremos";
    char *ptr;
    ptr = nota;
    puts(ptr);
    puts(++ptr);
    nota[7] = '\0' ;
    puts(nota);
    puts(++ptr);
    return 0;
}
```

9. Desarrolle un programa que lea e imprima el promedio de cada fila de una matriz de enteros de 3x5, utilizando una función que calcule el promedio de un vector de 5 enteros.

10. Declare una variable de tipo struct para almacenar nombre, matrícula y promedio de un alumno. Desarrolle una función que realice su lectura.

11. Desarrolle un programa que lea los datos de un conjunto de alumnos sobre un arreglo de structs y realice los siguientes procesos:

- ✓ Nombre del alumno con máximo promedio
- ✓ Listado, nombres, matrículas y promedio de los alumnos con promedio inferior a X (dato)

12. Preguntas Teóricas

- ¿Qué diferencia existe entre un vector de caracteres y una cadena de caracteres?
- ¿A que es equivalente `(char*)('\0')`?
- ¿Qué biblioteca se tiene que incluir para manejar cadenas de caracteres?
- ¿Las cadenas de caracteres se pueden asignar entre sí? Justifique su respuesta.
- ¿Por qué cuando se lee una cadena de caracteres con la función `scanf`, no se pasa la dirección de memoria de la misma?
- ¿Qué diferencias existen entre el `scanf` y el `gets`?
- Siendo `s` un arreglo de caracteres que almacena una cadena, ¿cuál es la diferencia entre `sizeof(s)` y `strlen(s)`?
- Indique al menos 3 funciones que transformen una cadena de caracteres en valores numéricos.
- Realice un cuadro indicando las diferencias en el uso de cadenas de caracteres en C y Pascal.
- Explicar qué pasa si se olvida el carácter nulo como último carácter de una cadena (string).
- ¿Cómo se visualizan las matrices en memoria?
- ¿Por qué es importante escribir la cantidad de columnas en la cabecera de una función o procedimiento a la que se le pasa como parámetro una matriz?
- Realice un cuadro comparativo acerca del uso de vectores y matrices entre C y Pascal.
- Investigue el uso de `typedef`. Aplique a la definición de tipo `struct`.

Ejercicios Adicionales

13. Compare los siguientes ciclos y explique su funcionamiento:

<p>a)</p> <pre>#define FIN "****" char dato[10]; gets(dato); while(strcmp(dato, FIN) != 0) { gets(dato); }</pre>	<p>b)</p> <pre>// debe tipear CTRL-Z + Enter // para finalizar char dato[10]; while(gets(dato) != NULL) { }</pre>
---	---

14. Desarrollar una función que utilice la función `cambio` para intercambiar de un arreglo de caracteres, los elementos simétricos (equidistantes al punto medio).

15. Indique qué realiza cada uno de los siguientes trozos de código:

<p>a) <code>char c[2][6] = {{ "hola"}, {"adios"}}</code></p>	<p>b) <code>int d[3][3] = {{1,2,3}, {2,3,4}, {5,6,7}}</code></p>
<p>c) <code>char c[2][2] = {48}</code></p>	<p>d) <code>int t[3][3] = {{1,2,3}, {4}, {5,6}}</code></p>
<p>e) <code>int a[3][3] = {1,2,3,4,5}</code></p>	<p>f)</p> <pre>int a[3][3], i, j; for (i = 0; i <= 2; i++) for (int j = 0; j <= 2; j++) a[i][j] = i + j;</pre>

16. Leer una matriz de `NxM` de enteros, generar un arreglo de structs que almacene para cada fila, el índice de la fila y

- la suma de los múltiplos de `N` (en dicha fila)
- si no hubiese múltiplos, la cantidad de pares (en dicha fila)

- si no hubiese ni múltiplos ni pares, no generar elemento para esa fila.

Ejemplo: Si $N=4$ y $M=3$

A

3	1	21
16	5	8
7	11	9
10	7	14

B

1	:	24	3	:	2
---	---	----	---	---	---

Nota: implementar funciones que evalúen los resultados pedidos en una fila de la matriz, pasando como argumento la fila como arreglo unidimensional.

17. En una estación de peaje hay 7 cabinas, los vehículos abonan el peaje de acuerdo a su categoría.

Al final del día se cuenta con un archivo de texto que contiene por cada auto que pasó por la estación:

- Cabina (1..7)
- Categoría (1..5)

Se pide, utilizando un arreglo que almacene el monto a pagar por cada categoría, leer el archivo para calcular e informar:

- Cantidad de autos en cada categoría
- Total recaudado en cada cabina
- Para la categoría más numerosa, ¿cuál fue la cabina que registró menos vehículos de dicha categoría?

18. Considere la misma problemática del ejercicio anterior, pero con los siguientes cambios:

- La cantidad de cabinas y categorías es N y M respectivamente.
- Las cabinas están identificadas con códigos de 4 caracteres y las categorías con códigos de 2 caracteres

Sugerencia: utilizar dos arreglos (tablas), uno con los códigos de cabinas y otro arreglo de structs con código de categorías y precios. A partir de la búsqueda del código obtener el índice (posición donde se encuentra). Evaluar la posibilidad que el código sea erróneo y no se encuentre.