



Programación 2

Objetivos

Desarrollar habilidades en la definición y utilización de estructuras estáticas de datos, como mecanismo de almacenamiento y programación para la manipulación de grandes volúmenes de información.

Utilizar un lenguaje de programación que permita la definición de las estructuras de datos estáticas tales como Arreglos (vectores y matrices), registros y vectores de registros.

Contenidos del curso

1. Introducción

1.1 La programación imperativa

1.2 Estructuras de control (repaso de conceptos)

1.3 Estructuras de datos estáticas (Conceptualización)

1.3.1 Arreglos (Unidimensionales o vectores, bidimensionales o matrices, multidimensionales)

1.3.2 Registros

1.3.3 Combinación de estructuras (Arreglos de registros)

1.4 Estructuras dinámicas de datos (conceptualización de listas, árboles, grafos)

1.5 Programación modular (la función como fundamento de la programación modular)

2. Estructuras de datos estáticas

2.1 Arreglos

2.1.1 Arreglos Unidimensionales o Vectores

2.1.2 Conceptualización

2.1.3 Definición de vectores en un lenguaje de programación

2.1.4 Captura, almacenamiento, recorridos y procesamiento de datos en vectores

2.1.5 Paso de vectores como parámetros a una función

2.1.6 Aplicación del concepto en la resolución de problemas

2.2 Cadenas de caracteres (String)

2.2.1 Las cadenas de caracteres como un caso particular de los vectores

2.2.2 Captura y almacenamiento de caracteres y de cadenas

2.2.3 Manipulación de vectores de caracteres (cadenas)

2.2.4 Funciones predefinidas para la manipulación de cadenas de caracteres

2.2.5 Aplicación del concepto en la resolución de problemas

2.3 Arreglos bidimensionales o Matrices

2.3.1 Conceptualización

2.3.2 Definición de Matrices en un lenguaje de programación

2.3.3 Captura, almacenamiento, recorridos y procesamiento de datos en Matrices

2.3.4 Paso de Matrices como parámetros a una función

2.3.5 Aplicación del concepto en la resolución de problemas

2.4 Registros

2.4.1 Conceptualización

2.4.2 Definición de registros en un lenguaje de programación

2.4.3 Captura, almacenamiento y procesamiento de datos en registros

2.4.4 Aplicación del concepto en la resolución de problemas

2.5 Combinación de estructuras estáticas de datos

2.5.1 Arreglos y matrices como campos dentro de un registro

2.5.2 Arreglos de registros

3. Archivos

3.1 Conceptualización

3.2 Entrada y salida de datos mediante archivos

3.3 Aplicación del concepto en la resolución de problemas

Bibliografía Recomendada

Fundamentos de Programación, Algoritmos, Estructuras de datos y Objetos, Luis Joyanes Aguilar, Tercera edición. Editorial McGraw Hill.

Informática Básica, Eduardo Alcalde y Miguel Garcia, Segunda edición, Editorial McGraw Hill.

Fundamentos de Programación, Algoritmos y Estructuras de datos, Luis Joyanes Aguilar, Segunda edición. Editorial McGraw Hill.

Fundamentos de Programación, Libro de problemas, Luis Joyanes Aguilar, Editorial McGraw Hill.

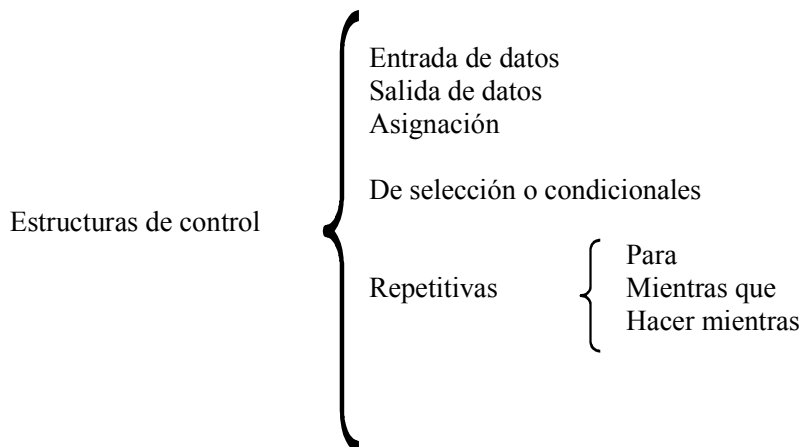
Turbo C/C++ 3.1 Manual de referencia, Herbert Schildt, McGraw Hill

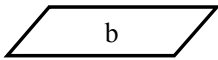
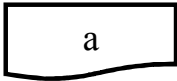
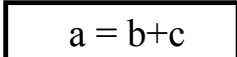


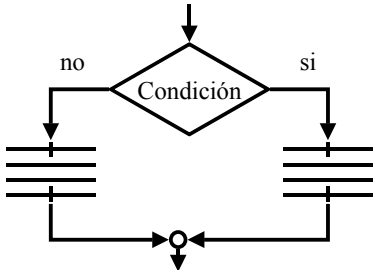
Contenidos de la Guía

1	RESUMEN DE NOTACIONES DE LAS ESTRUCTURAS DE CONTROL.....	3
2	ESTRUCTURAS DE DATOS	5
3	ARREGLOS UNIDIMENSIONALES (VECTORES)	5
4	VECTORES PARALELOS	11
5	CADENAS DE CARACTERES COMO CASO PARTICULAR DE ARREGLOS UNIDIMENSIONALES (VECTORES DE CARACTERES)	15
6	ARREGLOS BIDIMENSIONALES O MATRICES	18
7	REGISTROS.....	27
8	ARCHIVOS.....	42

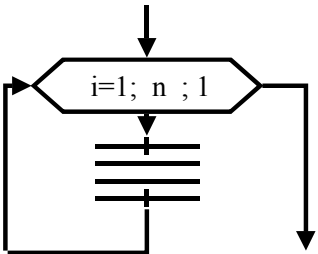
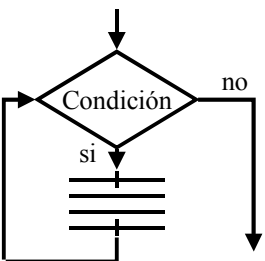
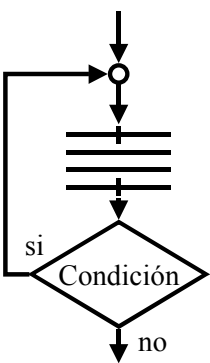
1 Resumen de notaciones de las estructuras de control



Estructuras de control			
Nombre	Diagrama de flujo	Seudocódigo	En C
Entrada o captura de datos		Leer b	<code>scanf ("%d", &b);</code>
Salida o impresión de datos		Imprimir a	<code>printf ("cadena de formateo ", a);</code>
Asignación		<code>a = b + c</code>	<code>a = b + c ;</code>

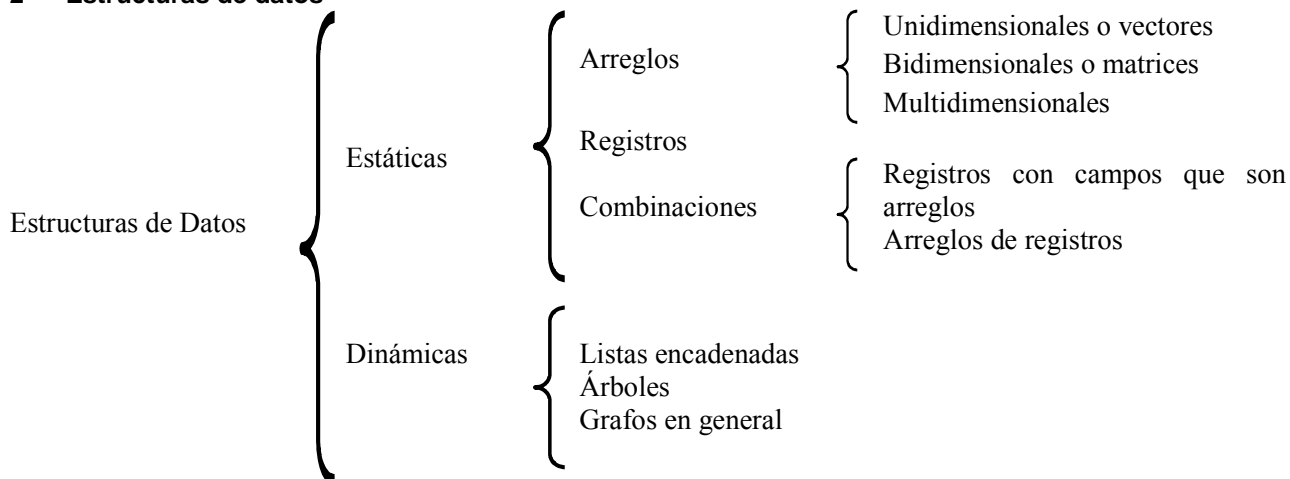
	<p>Si condición entonces Instrucciones para cuando la condición sea verdadera</p> <p>Sino Instrucciones para cuando la condición sea falsa</p> <p>Fin si</p>	<pre>if (condición) { Instrucciones en c para cuando la condición sea verdadera } else { Instrucciones en c para cuando la condición sea falsa };</pre>
---	--	---



<p>Estructura repetitiva Para</p>		<p>Para i=1 hasta n inc 1 haga Instrucciones a repetir fin para</p>	<pre>for (i =1 ; i<=n ; i=i+1){ Instrucciones a repetir };</pre>
<p>Estructura repetitiva Mientras que</p>		<p>MQ condición haga Instrucciones a repetir fin MQ</p>	<pre>while (condición) { Instrucciones a repetir };</pre>
<p>Estructura repetitiva Hacer Mientras</p>		<p>Hacer Instrucciones a repetir MQ condición</p>	<pre>Do { Instrucciones a repetir } while (condición);</pre>



2 Estructuras de datos



Se denominan estructuras estáticas aquellas colecciones de datos cuyo número de datos se define en tiempo de diseño, es decir cuando se está escribiendo el código del programa y por eso durante la ejecución del programa la colección de datos no podrá crecer en capacidad, es decir que si una estructura está diseñada para almacenar 50 datos, en tiempo de ejecución no podrá incrementarse este tamaño, por lo tanto solo podrá almacenar esos 50 datos.

Mientras que las estructuras dinámicas son colecciones de datos que en tiempo de ejecución pueden incrementar el número de miembros (datos) que pertenecen a la estructura. Este es el caso de las listas encadenadas para lo cual se requiere conceptos (que no son del alcance de esta materia) como la asignación dinámica de memoria.

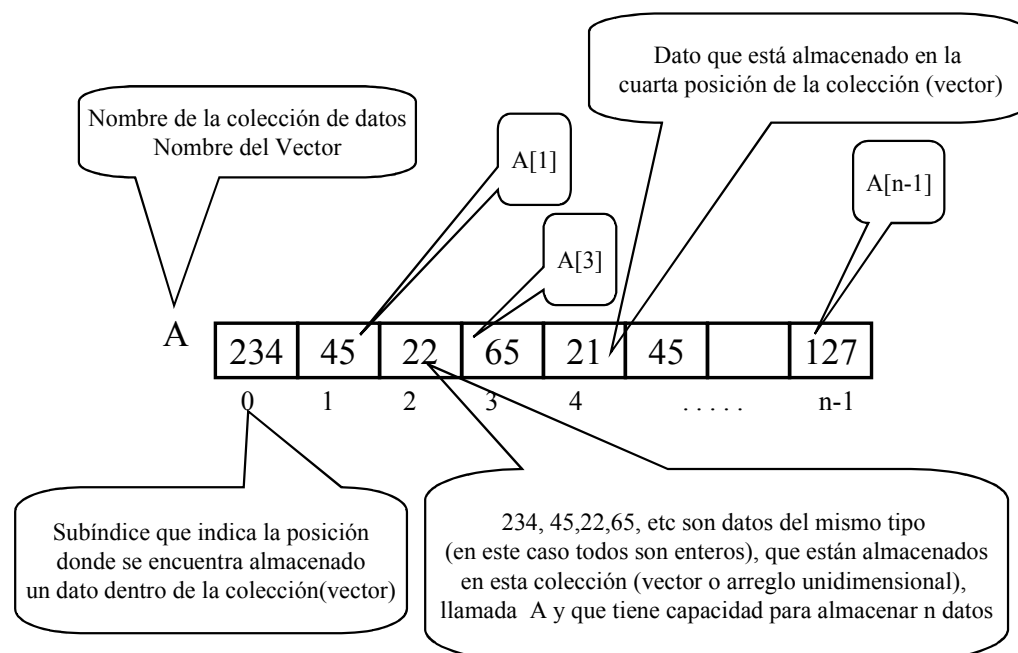
En algunos lenguajes se permite que en tiempo de ejecución se cambie el tamaño de algunas estructuras estáticas como los vectores...pero para evitar ambigüedad de conceptos, esto no se tendrá en cuenta. En este curso las estructuras estáticas no cambian de tamaño en tiempo de ejecución.

Dentro de las estructuras de datos estáticas están los arreglos que son colecciones de datos del mismo tipo, referenciadas con un solo nombre, y que para acceder a cada uno de los datos de la colección se hace uso de subíndices.

Cuando solo se necesita un subíndice para acceder a los datos de la colección, a estas estructuras se les llama arreglos unidimensionales o vectores. Si se necesitan dos subíndices se les llama arreglos bidimensionales o matrices y si es necesario tres o más subíndices se les conoce como arreglos multidimensionales.

3 Arreglos unidimensionales (vectores)

Un vector o arreglo unidimensional es una colección de datos del mismo tipo, referenciada con un solo nombre y que para acceder a cada uno de sus datos, se necesita un subíndice que en este caso indica la posición (de orden) donde se encuentra almacenado un dato, dentro de la colección.



Recuerde que todos los ejercicios se deben realizar de forma modular

1- Considere la función funcion1 y el arreglo G dado

Arreglo G

5	14	34	56	2	46	7	23	67	98	2	89
---	----	----	----	---	----	---	----	----	----	---	----

```
int funcion1(int x[], int n, int v){
    int r = 0, k = 6;
    for( i = 0; i<n; i++) {
        if( x[ i ] == v){
            r = i;
            k = k +1;
        }
    };
    return( x [ r ] );
};
```

	Pregunta	Rta
1	Diga cual es el valor de la variable r para la función funcion1, con el siguiente llamado a=funcion1(G,12,29); cuando el valor de i es 7.	
2	Diga el valor que retorna la función con el siguiente llamado a=funcion1(G,7,89);	
3	Cual debe ser el valor de p en el siguiente llamado a=funcion1(G,12,p); para que la función retorne 98	
4	Con que valor termina la variable k con el siguiente llamado a=funcion1(G,12,2);	
5	Con que valor queda a después de ejecutarse la siguiente asignación a = (funcion1(G,12,2)*10) - 5;	

2-Llenar un vector de 15 posiciones para posteriormente determinar el número menor y la posición que ocupa. (modif. A un vector de N posiciones “N dado por teclado”)

3-Llenar un vector de N posiciones (N dado por teclado) para posteriormente determinar el promedio de las posiciones pares y el producto de las impares.



4-Llenar dos vectores de N posiciones (N dado por teclado) para posteriormente, crear un tercer vector con la suma de los dos anteriores.

5-Llenar un vector de n posiciones para posteriormente, invertirlo sobre si mismo.

6-Llenar un vector de N posiciones (N dado por teclado) y posteriormente solicitar un numero “X” para determinar cuantas veces esta “X” repetido dentro del vector.

7-Llenar un vector de N posiciones (N dado por teclado) y posteriormente solicitar un numero “X” y un numero “Z” para reemplazar todos los valores “X” con ”Z” dentro del vector.

Insertar todos los valores de un vector V1 en un vector V2 a partir de una posición determinada en el vector V2 sin perder los valores de V2. (se debe Capturar por teclado el tamaño de los dos vectores)

8-Insertar todos los valores de un vector V1 en un vector V2 a partir de una posición determinada en el vector V2 sin perder los valores de V2. (se debe Capturar por teclado el tamaño de los dos vectores)

9-Hacer un programa en Lenguaje C para llenar un vector de N posiciones (N dado por teclado), para posteriormente crear un segundo vector con el espejo del primero.

	A	B (espejo)
0	12345	54321
1	589	985
2	4245	5424

n-1		

10-Llenar dos vectores A y B de n y m posiciones respectivamente (n y m dados por teclado). Supóngase que los vectores A y B representan conjuntos (hay que eliminar los repetidos de cada vector), por lo tanto hallar en vectores diferentes las siguientes operaciones entre conjuntos:

- A intersectado con B Valores del primero que están en el segundo.
- A unido con B Valores del primero y del segundo en un solo vector (sin repetir datos).
- A – B los valores que están en A y que no están en B
- A diferencia simétrica con B, son los que están en A que no están en B y los que están en B que no están en A

11- Capturar por teclado los datos de un vector de enteros de n datos (n dado por teclado y $1 < n < 200$) y clasificar los datos del vector en dos vectores uno con los números primos y otro con los números perfectos.

Un número es primo si solo tiene dos divisores, 1 y él mismo, ejemplos 1,2,3,5,7,11 etc.

Un número es perfecto si la suma de sus divisores propios es igual a él mismo. Ejemplo el 6,28, etc.

Por ejemplo para un vector dado por teclado como el siguiente

2	40	25	7	40	28	17	45	6	0	4	5
---	----	----	---	----	----	----	----	---	---	---	---

Los datos clasificados en el vector de primos es:

2	7	17	5
---	---	----	---

Y el vector de perfectos es

28	6
----	---

12-En la piscina de la universidad se va a realizar un campeonato de clavados para n deportistas (n dado por teclado). Los deportistas serán calificados por un jurado de 10 personas. Para calcular su puntuación se determina el promedio de los puntajes dados por jurado calificador pero con la única salvedad que la peor y la mejor nota no se tienen en cuenta.

Cree un vector con el promedio de las calificaciones y luego determine el promedio mas alto y la mas bajo de los deportistas

Ejemplo:

4.9	7.6	5.5	3.1	7.9	4.6	9.0	8.4	5.4	6.2
0	1	2	3	4	5	6	7	8	9

Prom= $(4.9+7.6+5.5+7.9+4.6+8.4+5.4+6.2)/8$

Las calificaciones 3,1 y 9,0 no son tenidas en cuenta por ser la más alta y la más baja



13-Llenar un vector A de n posiciones (n dado por teclado), para posteriormente, ordenarlo de mayor a menor en otro vector y de menor a mayor en el mismo vector A. Utilice el método de burbuja y averigüe por otros métodos de ordenación que se podría utilizar para realizar la misma tarea.

14-Lea dos vectores de igual tamaño y cree un nuevo vector con los datos de los dos primeros pero intercalados. Por ejemplo si los dos vectores son:

2	40	25	7	40	28	17
---	----	----	---	----	----	----

3	45	23	2	25	45	72
---	----	----	---	----	----	----

El vector resultante es:

2	3	40	45	25	23	7	2	40	25	28	45	17	72
---	---	----	----	----	----	---	---	----	----	----	----	----	----

15-Se tiene el arreglo

ventas

0	1	2		10	11
V _{ene}	V _{feb}	V _{mar}	· ·	V _{nov}	V _{dic}

Donde se almacenan las ventas mensuales de una empresa.

Elabore una función que muestre los siguientes datos:

¿En qué mes se dieron las ventas máximas de la empresa?

¿A cuánto ascendieron las ventas máximas?

¿Cuál fue el total de las ventas?

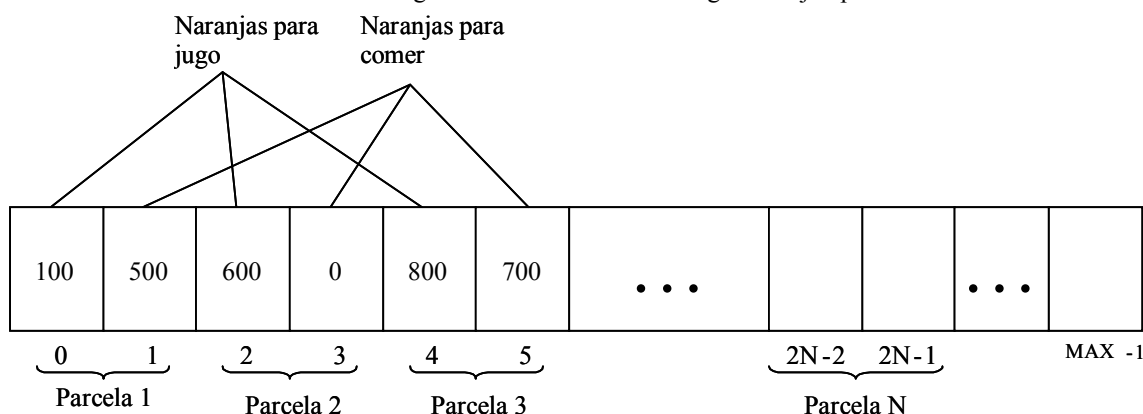
¿Cuál fue el promedio de las ventas?

16-Se tiene un conjunto de n parejas de datos X_i, Y_i donde cada pareja representa las coordenadas del punto i de un polígono irregular de n lados. Suponga que las coordenadas se proporcionarán en orden adyacente. Elabore una función para Capturar por teclado el número de lados del polígono, Capturar por teclado las coordenadas de los puntos y obtener el área del polígono irregular utilizando la siguiente fórmula:

$$\text{Area} = [(X_0 + X_1)(Y_0 - Y_1) + (X_1 + X_2)(Y_1 - Y_2) + \dots + (X_{n-1} + X_0)(Y_{n-1} - Y_0)]/2$$

Suponga que los X_i, Y_i son reales y n es entero con $1 \leq n \leq 50$.

17-Una cooperativa de productores de naranjas almacena el total de toneladas cosechadas durante el último año en N parcelas ($1 \leq N \leq 50$). En cada parcela se pueden cultivar dos tipos de naranjas: para jugo y para comer. Se conoce el total de toneladas cosechadas de cada uno de los tipos de naranjas. Si en una parcela no se hubiera cosechado alguno de los tipos, entonces habrá 0. La información se almacena en un arreglo como se muestra en el siguiente ejemplo:



En la parcela 1 se cosecharon: 100 toneladas de naranjas para jugo y 500 toneladas de naranjas para comer.

En la parcela 2 se cosecharon: 600 toneladas de naranjas para jugo y 0 toneladas de naranjas para comer.

NOTA: Observe que la información de una misma parcela ocupa posiciones consecutivas en el arreglo.

Se le pide que analice, diseñe y codifique un programa, usando módulos, que pueda:

Capturar por teclado la información: N ($1 \leq N \leq 50$) y las toneladas por tipo de naranja de cada una las parcelas.



Calcular e imprimir el total de la producción por parcela.

Eliminar la información de una parcela. El dato requerido para esta opción es el número de parcela a eliminar.

Buscar e imprimir el número de una parcela (si hubiera) que no haya tenido producción de ninguno de los tipos de naranjas. Es decir, durante el último año su producción total fue 0.

18-Hacer un programa en Lenguaje C para capturar por teclado los datos de dos vectores de 12 posiciones denominados prod1 y prod2. Dichos vectores almacenarán la cantidad de unidades generadas de cada producto, mes a mes. Adicionalmente, capturar por teclado los datos de un tercer vector denominado costou de dos posiciones, el cual almacenara el costo unitario de los dos productos.

- Crear una función que permita: en un cuarto vector denominado costot de 12 posiciones, calcular y almacenar el costo de la producción total, mes a mes, así:
- $\text{Costo Total} = (\text{Cantidad Unidades del Prod1} * \text{Valor Unitario Prod1}) + (\text{Cantidad Unidades del Prod2} * \text{Valor Unitario Prod2})$
- Crear una función que calcule y retorne el promedio de cada uno de los cuatro trimestres del vector costot.
- Crear una función que calcule y retorne el mes, donde se encuentra el mayor valor del vector costot. (si existen dos iguales, se asume como mayor el primero)

19-Dado un vector V1 de tipo entero de tamaño n, donde n se da por teclado, hacer un programa que:

Llene el vector V1 a través de una función

Invierta el vector V1 sin utilizar otro vector (en el mismo)

Genere un vector V2 que será el contenido de las posiciones impares de V1 ya invertido a través de una función.

Imprima el vector V2 por medio de una función imprimir en forma ascendente.

Ejemplo:

V1

5	7	1	3	60	30
0	1	2	3	4	5

V1 invertido

30	60	3	1	7	5
0	1	2	3	4	5

V2

60	1	5
0	1	2

20-Dados 2 vectores V1 y V2 de tipo entero de tamaño n y m, donde n y m se da por teclado, hacer un programa que:

Llene los vectores V1 y V2 con una función llenar.

Genere un vector V3 que es el resultado de concatenar (Unir) V1 y V2 a través de una función

Genere un vector V4 a partir de V3, donde V4 contiene aquellos números perfectos que existen en V3.

Nota: Un número perfecto es aquel número tal que la suma de sus divisores propios es igual al mismo número. Y los divisores propios de un número son todos sus divisores excepto el mismo número. Ejemplo.

V1

4	6	12	28	60	5
0	1	2	3	4	5

V2

2	45	496
0	1	2

V3

4	6	12	28	60	5	2	45	496
0	1	2	3	4	5	6	7	8

V4

6	28	496
0	1	2

21-Crear un programa en lenguaje C que registre en un vector de enteros muchos números (máximo 1000 números), hasta finalizar con cero. Sólo se podrán ingresar los números 1, 2 y 3 (en cualquier orden). Se pide contar la cantidad de veces que se ingreso la



secuencia: 1 2 3. Por Ejemplo: Si se ingresaron los siguientes datos: 1,1,2,2,1, 2, 3, 1, 1, 1, 1, 3, 1, 1, 2, 1, 1, 1, 3, 3, 3, 1, 2, 3, 2, 2. Las veces que se ingreso la secuencia 1,2,3 es 2 veces

Realizar las Funciones:

- Main
- Llenar el vector
- Contar la cantidad de veces que se ingreso la serie 1 2 3

22- Cree un programa en C que lea un vector de N posiciones (N dado por teclado) y posteriormente cree un segundo vector con todos los valores que se repitan exactamente 2 veces en el primer vector imprima el vector resultante. Realice:

- Main
- Lectura
- Creación del segundo Vector
- Impresión del segundo vector

23- Cree un programa en C que lea un vector de N posiciones (N dado por teclado) y posteriormente cree un segundo vector con los valores de la siguiente forma:

1er Mayor	1er Menor	2do Mayor	2do Menor	3er Mayor	3er Menor
0	1	2	3	4	5		n-1

Realice:

- Main
- Lectura
- Creación del 2 Vector
- Impresión del 2 vector

24- Haga un Programa en C para llenar un vector A de N posiciones (**N Máximo 50**) “**N DESCONOCIDO**” **¡No Puede ser capturado por teclado!** La lectura de los Valores termina cuando el usuario ingrese un número que ya está almacenado en el vector o cuando el vector llegue a su máximo de posiciones.

Cree un nuevo Vector B, ordenado de menor a mayor (a partir del vector A).

Cree un Vector C con los valores Impares y Pares intercalados (en ese Orden) del Vector B y en el momento en que no se puedan intercalar terminaría el llenado del Vector C.

Ejemplo:

Se llena el vector A.

A

6	2	18	12	7	4	9	1
---	---	----	----	---	---	---	---

Se crea el Vector B ordenando su contenido de menor a mayor

B

1	2	4	6	7	9	12	18
---	---	---	---	---	---	----	----

Se crea el Vector C con los valores intercalados Impares/Pares del Vector B

C

1	2	7	4	9	6
---	---	---	---	---	---

Cree:

- Main.
- Función para el Llenado del **Vector A**.
- Función para la Creación y Ordenamiento del **Vector B**.
- Función para la Creación del **Vector C**.
- Función para la impresión de cualquier vector. Debe usarse en el main para imprimir los tres vectores

Nota: Puede usar vectores auxiliares si lo cree necesario



25-Elabore un programa en lenguaje C que lea dos arreglos de números enteros (entre 1 y 100) A y B de cualquier tamaño (no necesariamente de igual tamaño) (el tamaño debe ser de máximo 500 posiciones cada uno) y luego construya un tercer vector C, que contenga los datos almacenados en los arreglos A y B en forma intercalada de a dos elementos. Es decir, dos datos del arreglo A y luego dos datos del arreglo B, así sucesivamente. Si se terminan los datos del arreglo A y aun quedan datos en B, se deben pasar todos los datos que sobran de B al arreglo C en forma consecutiva. Y en caso que se terminen primero los datos del arreglo B y aun queden datos en A se deben pasar todos los datos que sobran de A al arreglo C en forma consecutiva.

Por ejemplo sea A:

5	12	4	9	3
0	1	2	3	4

y sea B:

2	11	7	22	45	1	16	8	14
0	1	2	3	4	5	6	7	8

Entonces se debe construir un arreglo C con los siguientes valores:

5	12	2	11	4	9	7	22	3	45	1	16	8	14
0	1	2	3	4	5	6	7	8	9	10	11	12	13

El programa debe contener funciones que realicen lo siguiente:

Validar la lectura de números enteros en el intervalo de 1 a 100. (valor 1,0)

Llenar arreglos de cualquier tamaño.(valor 1,0)

Construir el vector C en la forma indicada en el enunciado anterior, recibiendo como entrada los arreglos A y B llenos.(valor 2,0)

Función main.(valor 1,0)

26-Elabore un programa en lenguaje C capture por teclado los elementos de un vector T (de máximo 100 posiciones) de enteros positivos, sin dar por teclado el tamaño del vector, hasta que se capture un entero negativo (este elemento negativo no hará parte del vector). Después construir un vector S a partir de los datos del vector T así:

$S(0)=T(0)$

$S(1)=T(0) - T(1)$

$S(2)=T(0)+T(1)+T(2)$

$S(3)=T(0) - T(1) + T(2) - T(3)$

$S(4)=T(0) + T(1) + T(2) + T(3) + T(4)$

$S(5)=T(0) - T(1) + T(2) - T(3) + T(4) - T(5)$

$S(k)= \dots\dots\dots$

Nótese que para las posiciones pares de S se realizan solo suma de elementos de T, pero para las posiciones impares de S se realizan sumas y restas intercaladas de elementos de T.

Elaborar las siguientes funciones

Función de llenado como se indica en el enunciado

Fundón para construir el vector S a partir de los elementos del Vector T

Función main

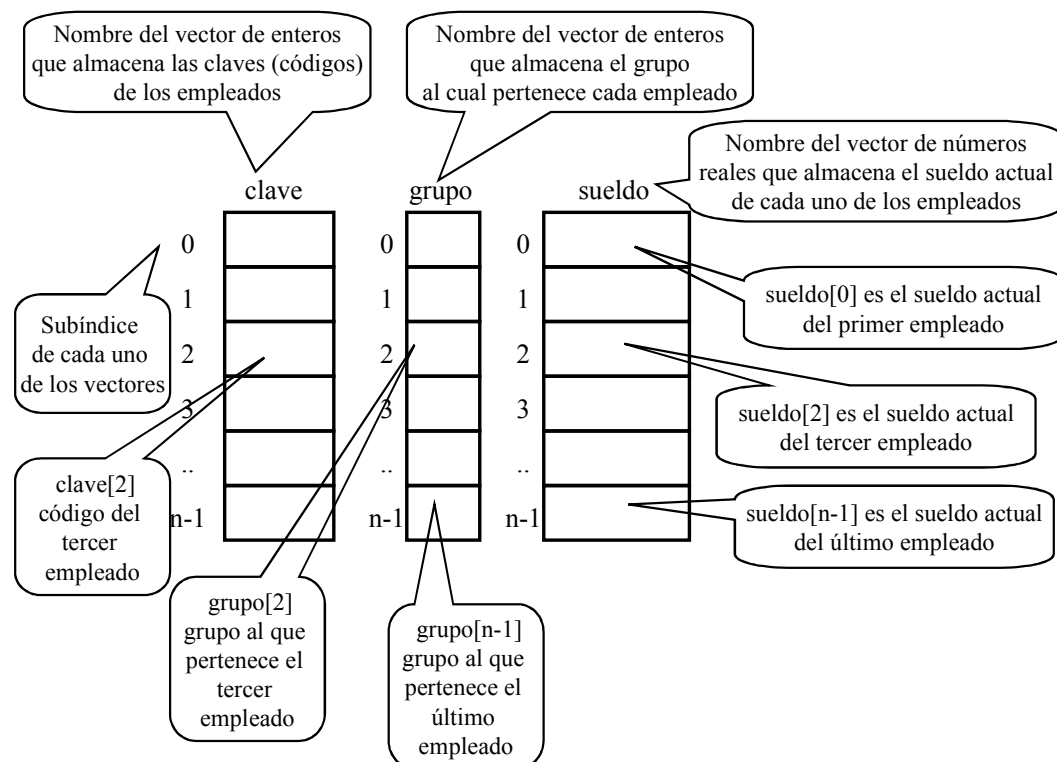
4 Vectores paralelos

27-Una compañía ha clasificado a sus n ($1 \leq n \leq 50$) empleados dentro de los grupos 1 y 2. Cada empleado del grupo 1 recibirá, a partir de ahora, un aumento de \$250.00 a la semana sobre su sueldo actual, mientras que a cada uno de los del grupo 2 sólo se le dará \$150.00 adicionales por semana. La información que se proporciona de cada empleado es: clave del empleado (comprendida entre 100 y 2500, inclusive), grupo al que pertenece (1 o 2) y sueldo semanal actual (entre \$300.00 y \$6000.00). Si un empleado del grupo 1 con el nuevo sueldo gana semanalmente más que el sueldo promedio del grupo, entonces debe cambiar al grupo 2.

Analice, diseñe y codifique un programa modular que:

Lea la información de los n empleados e imprima la clave, grupo y sueldo semanal nuevo para cada empleado, actualizando el grupo si es el caso.

Calcule el sueldo promedio de los empleados por grupo y calcule también el promedio general de todos, sin considerar el grupo. Indique cuántos empleados cambiaron del grupo 1 al 2.



28-Llenar 2 vectores V1 y V2 de N posiciones (N dado por teclado), para posteriormente, ordenarlo de mayor a menor de forma paralela usando a V1 como vector base

29-Un casino desea llevar a cabo unas estadísticas sobre el juego de ruleta. Se solicita un programa modular que cuente el número de veces que cada número es ganador en una jornada y al final del día imprima un reporte con lo siguiente:

¿Cuántas veces resultó ganador cada elemento de la ruleta?

¿Cuáles elementos no resultaron ganadores en toda la jornada?

¿Cuál fue el número de cada color que ganó más veces?(suponga que sólo hay uno)

¿Cuál fue el elemento que ganó en más ocasiones?

Recuerde que la ruleta está formada por 70 elementos. Existen del 1 al 34 rojos, del 1 al 34 negros, el 0 y el 00. El casino desconoce el número de juegos de una jornada. Los datos que el casino proporcionará son los siguientes:

número, color

número, color

...

-1

Donde:

Número representa el número correspondiente al elemento ganador, con $0 \leq \text{número} \leq 34$

Donde el 0 se representa como un 0 rojo y el 00 se representa como un 0 negro.

Color representa el color correspondiente al elemento ganador con: R = rojo, N = negro

-1 representan el fin de la jornada

30-Se lee la información de 100 estudiantes, en tres vectores

En el vector A se almacena LA NOTA DEL ESTUDIANTE

En el vector B se almacena LA CARRERA

En el vector C se almacena EL CODIGO del estudiante



Existen las siguientes carreras adscritas cada una a una facultad.

Facultad	Carrera
Ingenierías	1. Ing Sistemas
Ingenierías	2. Ing Industrial
Ingenierías	3. Ing Mecatronica
Ciencias Económicas	4. Contaduría
Ciencias Económicas	5. Admón. empresas
Ciencias Económicas	6. Economía
Salud	7. Medicina
Salud	8. Nutrición

Se debe calcular e imprimir

Cuántos estudiantes hay en cada facultad

El código del estudiante con la nota más alta en cada Facultad

El promedio de notas de los estudiantes por carrera

31- Se desea registrar la información de código y temperatura de 50 ciudades Realizar funciones para:

Capturar por teclado los datos de los vectores

Capturar por teclado la temperatura donde desea vivir una persona hasta cuando se digite 0, e imprimir los posibles códigos de las ciudades que tienen dicha temperatura, en caso contrario imprimir “No existe ciudad”.

Calcular e imprimir el código de la ciudad con mayor temperatura.

Ordenar los dos vectores teniendo en cuenta la temperatura en orden ascendente. Nota: los dos vectores deben ser ordenados al tiempo de forma paralela

32- De los 100 alumnos de programación I, se desea almacenar la siguiente información, para cada uno de ellos, en tres vectores que corresponden a la siguiente información: Código, Edad, Zona (Los datos serán [1. Urbano 2. Rural])

La zona permite definir el valor de matrícula así:

Zona	Valor
2	500.000.00
1	1.000.000.00

El valor de matrícula tendrá un descuento, del 30% solo para los estudiantes menores de edad. (un menor de edad es el estudiante que tiene menos de 18 años)

Realizar: (implementando funciones)

Capturar por teclado la información de los estudiantes en los vectores (código, edad, zona)

Calcular e imprimir el valor de la matrícula de cada estudiante

Contar los estudiantes menores de edad, que residen en la zona Urbana y Rural

33- Una empresa de flores (rosas) desea llevar el control de sus ventas en el mes de la mujer cuenta con la siguiente información Código del tipo de rosa, Cantidad cultivada, Cantidad Vendida y precio de venta. Para N tipos de rosas (N dado por teclado).

Realice las siguientes funciones:

Llenado de la información.

El código y la cantidad en unidades del producto más vendido.

El código y la cantidad en unidades del producto menos producido.

El total de las ventas.

34- Un Banco de la ciudad quiere que usted cree un programa modular en C para llevar el control de los Números de Cuenta, Cédulas de Clientes y Saldo de la Cuenta (un cliente puede tener varias cuentas). En el Banco Existen 325 cuentas.

Debe realizar el llenado de la información Números de Cuenta, Cédulas de Clientes y Saldo de la Cuenta, posteriormente debe solicitar el número de cédula de un cliente para indicarle el saldo total de las cuentas del cliente en el banco, debe indicar el cliente con menos dinero en el banco y por ultimo debe solicitar Dos numero de cuenta (Cuenta Origen y Cuenta Destino) y un monto el cual será restado de la cuenta Origen y sumado a la cuenta Destino. Realice:

- Main
- Función Llenado
- Función Saldo por Cliente
- Función Cliente con menos dinero
- Función Transferencia



Nota: Para la función transferencia debe validar que la cuenta Origen tenga fondos suficientes para poder realizar la transferencia.

35-El instituto de bienestar familiar necesita manejar la información de adultos mayores beneficiados en uno de sus programas sociales. Para ello requiere manejar la siguiente información: documento de identidad, edad, estatura (en metros), peso (kilogramos). Se debe calcular el índice de masa corporal para cada individuo mediante la formula $IMC = \text{masa(kg)} / \text{estatura}^2(\text{metros})$. Si el índice de masa esta por debajo de 21, se dice que hay desnutrición, si está entre 21 y 25, está en su peso normal y por encima de 25 hay sobrepeso. Realizar un programa modular que me permita.

- Llenar la información de todos los individuos.
- Determinar el índice de masa corporal para cada individuo
- Determinar la cantidad de individuos con sobrepeso, con peso normal y con desnutrición.
- Determinar el promedio de edades de los individuos, así como el mayor de todos y el más joven de todos.

36-En un Instituto de educación existen 10 estudiantes, cada uno ve 3 materias (1001 Matemática, 1002 Biología y 1003 Dibujo) y en cada una de ellas se almacena la nota que saco. Cree 3 vectores un vector de Código del Estudiante, un Vector de Código de Materia y un vector de Notas. Ejemplo:

Código estudiante	Código Materia	Nota Materia
2116	1003	4,2
2133	1001	2,4
2116	1002	3,5
2436	1003	1,2
2116	1001	5,0
2436	1003	2,5

- Función para llenar la información
- Función para determinar el promedio general por materia
- Función para determinar el promedio de un estudiante (dado su código por teclado)
- Función para determinar la nota más alta por materia

37- En una empresa se piensa contratar a máximo 20 empleados, de los cuales se necesita saber la cédula, edad y salario a que aspira, datos que serán almacenados en tres vectores paralelos. Dada la cantidad de candidatos a contratar por teclado (no sobrepasar de 20), introducir los datos de cada uno. Si el candidato es menor a 30 años y aspira un salario entre 500.000 y 800.000 este se puede contratar. Se debe tener en cuenta que quienes reciban un salario menor a 600.000 tendrán una bonificación de 5% de su salario. Haga un programa en C con uso de funciones que permita al final saber:

- Cuantos pueden ser contratados según el salario a que aspiran
- El total de la nómina que debe pagar la empresa, según los posibles contratados.
- Cuantos de los posibles contratados tienen bonificación
- Cual de los posibles contratados tiene mayor edad.
- En el main además debe imprimir la información de cada uno de los posibles contratados (cédula, edad, salario) con su correspondiente bonificación

38- En una fábrica de textiles se almacena la información de las telas existentes en tres arreglos unidimensionales de la siguiente manera:

CÓDIGO	METROS	COLOR

El código de la tela es un número de 5 dígitos que se encuentra estructurado de la siguiente manera:

Los dos primeros dígitos (de izquierda a derecha) identifican el tipo de tela:

11 seda, 22 dacron, 33 algodón, 44 paño, 55 algodón

El tercer dígito (de izquierda a derecha) permite identificar el diseño de la tela

1 unicolor, 2 líneas, 3 cuadros, 4 rombos.

Los dos últimos dígitos corresponden al consecutivo de la tela, por regla este número debe encontrarse entre 11 y 88.

Los colores usados en la fábrica se identifican con un número de un solo dígito, y son los siguientes:

1 Rojo, 2 Verde, 3 Azul, 4 Amarillo, 5 Naranja, 6 Violeta, 7 Blanco, 8 Negro, 9 Gris.

Realice un programa en C++ que implemente las siguientes funciones:

Función validar: recibe un número (correspondiente a un código) y retorna 1 si el código es válido, 0 si no lo es.

Función llenado: Recibe los tres arreglos, y la cantidad de registros a llenar, solicita los datos (código, metros y color) y almacena en cada arreglo la información correspondiente. Implementa la función validar, para validar el código ingresado, en caso de ser incorrecto deberá solicitar nuevamente el ingreso del mismo. Para el color, en caso de capturar por teclado un número de más de un dígito deberá almacenar el último del mismo (ej. 1428 almacena el 8).

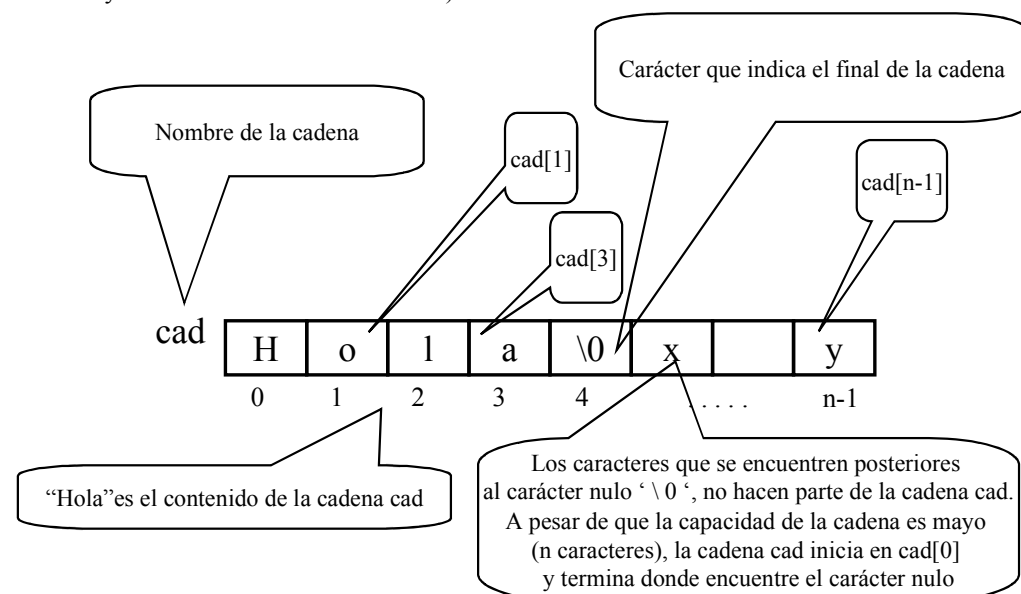
Función buscar tela: la función recibe los tres arreglos y tres variables que identifican un tipo de tela, un diseño y un color, la función debe retornar la cantidad de metros de tela que existen con esas especificaciones (tipo de tela, diseño y color).

Función buscar color: la función recibe como entrada los tres arreglos y un número que identifica un color a buscar, la función debe imprimir en pantalla los códigos de las telas que son del color señalado, y retornar el código de esas telas que tiene mayor metraje.

Función Main: Implementa las funciones anteriores, solicitando al usuario los datos que serán enviados como argumentos a cada una de las funciones.

5 Cadenas de caracteres como caso particular de arreglos unidimensionales (vectores de caracteres)

Una cadena de caracteres no es más que un arreglo unidimensional (vector) de caracteres, es decir que en cada casilla del vector se hay un carácter. A diferencia de otros tipos de vectores, las cadenas se manipulan sabiendo que para toda cada existe un carácter dentro de la cadena que indica el final de dicha cadena, este carácter se conoce como el carácter nulo y su valor es representado como ‘\0’, es así como se determina donde inicia una cadena (en la casilla 0) y donde termina (hasta encontrar una casilla del vector cuyo contenido sea el carácter nulo).



Recuerde que todos los ejercicios se deben realizar de forma modular y si no se indica se asume que las cadenas son de max 50 caracteres

39- Capturar por teclado una cadena de caracteres, determine la longitud de esta y escribirla al revés.

40- Capturar por teclado una frase de a lo más, 50 caracteres, cuente cuántas veces aparece la letra ‘a’ en la frase. Cuente cada una de las vocales minúsculas o mayúsculas que contiene. Realice una copia de esta en otra variable, Convertir la cadena (copia) a mayúsculas (con otra función a minúsculas).



41- Capturar por teclado una cadena de caracteres y la cortarla en palabras escribiendo cada palabra en una línea. Se supone que las palabras se separan sólo por blancos.

42- Capturar por teclado una frase de a lo más, 50 caracteres y cree una segunda cadena con la primera palabra de la frase original que empiece con la letra 'b', si existe.

43- Capturar por teclado dos cadenas de caracteres en forma separada y luego las una, en una nueva cadena pero intercalando palabra por palabra entre las dos cadenas. Por ejemplo si las cadenas son:

todos deportistas conocían prefieren creyendo milagros

los lo pero seguir en

Su programa debe escribir: todos los deportistas lo conocían pero prefieren seguir creyendo en milagros

44- Capturar por teclado una cadena de caracteres y llamar a un función que la modifique de la siguiente manera: se deben cambiar mayúsculas por minúsculas, minúsculas por mayúsculas y dígitos por puntos.

45- Capturar por teclado una cadena de caracteres, e indicar si la cadena leída es o no un palíndromo (se lee igual de izquierda a derecha, que de derecha a izquierda sin tener en cuenta los espacios). Ej: Dabale arroz a la zorra el abad

46- Decodificar una sucesión de a's y b's que aparece codificada por medio de un conjunto de parejas de dígitos y letras donde el dígito indica el número de veces que se repite la letra que lo sucede. El programa debe Capturar por teclado en una línea la sucesión codificada para después decodificarla. Si la sucesión codificada leída es: 3a4b2a su programa debe escribir: aaabbbbaa. Tener en cuenta que:

- Los números son de un sólo dígito (no hay 10,11, ...).
- Puede haber tantas sucesiones de a's y b's como se desee.
- Las letras no necesariamente vienen alternadas.
- En la línea de lectura, no hay ningún espacio.
- Considere que la sucesión decodificada será de longitud menor o igual a 80 caracteres.

47- Capturar por teclado una frase y una palabra y que cuente cuántas veces aparece la palabra leída dentro de la frase, ya sea como una palabra completa o como parte de otra palabra. La frase y la palabra tienen una longitud máxima de 50 caracteres.

48- El teclado de su computadora está descompuesto y cada vez que oprime la letra 's' ésta se imprime tres veces en vez de una. Hacer un programa en Lenguaje C que reciba como dato de entrada una línea tecleada en su computadora y que genere e imprima una cadena resultado sin exceso de 's'. El programa también debe funcionar si la línea original no contiene ninguna 's'. Si existen 6 's' consecutivas significa que el texto original tenía dos "ss" consecutivas

49- Plataforma envía sus mensajes codificados para que los intrusos no los descubra. Haga un programa para Plataforma que reciba un texto (máximo 50 caracteres) codificado, genere una cadena de caracteres con el texto decodificado y la imprima.

El tipo de codificación que utiliza Plataforma para escribir sus mensajes consiste en sustituir las vocales por números, dejando los demás caracteres igual, de la siguiente forma:

A → 1
E → 2
I → 3
O → 4
U → 5

Ejemplo:

Texto codificado: T4m1 1ñ4s c4nstr53r 1l c4nf31nzt1 y s4l4 m3n5t4s plr1 d2str53rl1.

Texto decodificado: Toma años construir la confianza y solo minutos para destruirla

50- Haga un programa que lea n parejas de palabras, las cuales son antónimas. El programa debe escribir las letras comunes a las dos palabras. Si una letra aparece más de una vez en la palabra y en su antónimo, debe registrarse una sola vez. Al final se debe imprimir el número del par que tiene más letras en común y la cantidad de letras en común:

Por ejemplo:

interesante	aburrido	ira
gordo	flaco	o



alegre triste er
alto bajo ao
Par con más letras en común: es el 1 y tiene 3 letras en común

51- Capturar por teclado una cadena de caracteres de máximo 50 caracteres y cree dos nuevas cadenas con las palabras en mayúsculas y minúsculas (Cada palabra en la cadena esta escrita en su totalidad en mayúscula o en minúscula) Ejemplo:

A

E	L		p	e	r	r	o		E	S		G	R	A	N	D	E	\0		
---	---	--	---	---	---	---	---	--	---	---	--	---	---	---	---	---	---	----	--	--

Cadenas Resultantes:

B

E	L		E	S		G	R	A	N	D	E	\0		
---	---	--	---	---	--	---	---	---	---	---	---	----	--	--

C

p	e	r	r	o	\0		
---	---	---	---	---	----	--	--

52- Capturar por teclado una cadena de caracteres de máximo 50 caracteres y posteriormente convertir el principio y fin de cada palabra en su equivalente en mayúscula. Ejemplo:

A

L	a		c	a	s	a		e	S		b	o	n	i	t	a		y		G	r	a	n	d	e	\0
---	---	--	---	---	---	---	--	---	---	--	---	---	---	---	---	---	--	---	--	---	---	---	---	---	---	----

Cadena Resultante:

A

L	A		C	a	s	A		E	S		B	o	n	i	t	A		Y		G	r	a	n	d	E	\0
---	---	--	---	---	---	---	--	---	---	--	---	---	---	---	---	---	--	---	--	---	---	---	---	---	---	----

Recuerde la relación entre las mayúsculas y minúsculas están dadas por la estructura de la Tabla de códigos Ascii:

A	65	a	97
B	66	b	98
C	67	c	99
.....		
Z	90	z	122

53-Dadas tres frases (cadenas de caracteres) así: la primera frase contiene un conjunto de palabras en ingles que representa un diccionario de palabras en ingles. La segunda frase contiene un conjunto de palabras en español que representan la traducción de las palabras en ingles contenidas en la primera frase. La tercera frase contiene palabras en ingles que se desean traducir a español. Ejemplo:

Primera frase: yes you this open file when friends hello

Segunda frase: si usted este abrir archivo cuando amigos hola

Tercera frase: hello friends

Resultado: hola amigos

Haga un programa que: Lea las tres frases, a través de una función, traduzca la tercera frase e imprima su contenido en español.

NOTA: por cada palabra en ingles de la primera frase existe una palabra en español en la segunda frase, en la misma posición como se muestra en el ejemplo y, las palabras de la tercera frase son del mismo tamaño que las de la primera frase.

54-Una cadena es un anagrama de otra, si: Ignorando todos los caracteres salvo las letras y tratando como iguales las letras mayúsculas y minúsculas, contienen las mismas letras (no necesariamente en el mismo orden). Escriba un programa que contenga las siguientes funciones:

- main: Donde se leen las dos cadenas (en este caso se asume que todas las letras son minúsculas), y se hace el llamado de la función anagrama.
- anagrama: Que reciba dos cadenas e inserte en ambas cadenas los caracteres necesarios para que cada una pase a ser un anagrama de la otra.

Ejemplo:

Antes de la llamada: cad0="gato", cad1="raton"

Después de la llamada: cad0="gatorn", cad1="ratong"

55-Una empresa de seguridad ha decidido implementar una estrategia de cifrado de mensajes. Para ello realiza el siguiente procedimiento. Dado un mensaje (cadena de hasta 200 caracteres), primero elimina los espacios en blanco, luego pasa el mensaje completamente a mayúsculas y después reemplaza cada letra por la que tiene el código ASCII más 10, Dado que el código ASCII de la 'Z' es el 90, si al sumar 10 al código de un carácter pasa de 90, el valor que excede se suma a 64 (ya que el código de la 'A' es el 65).

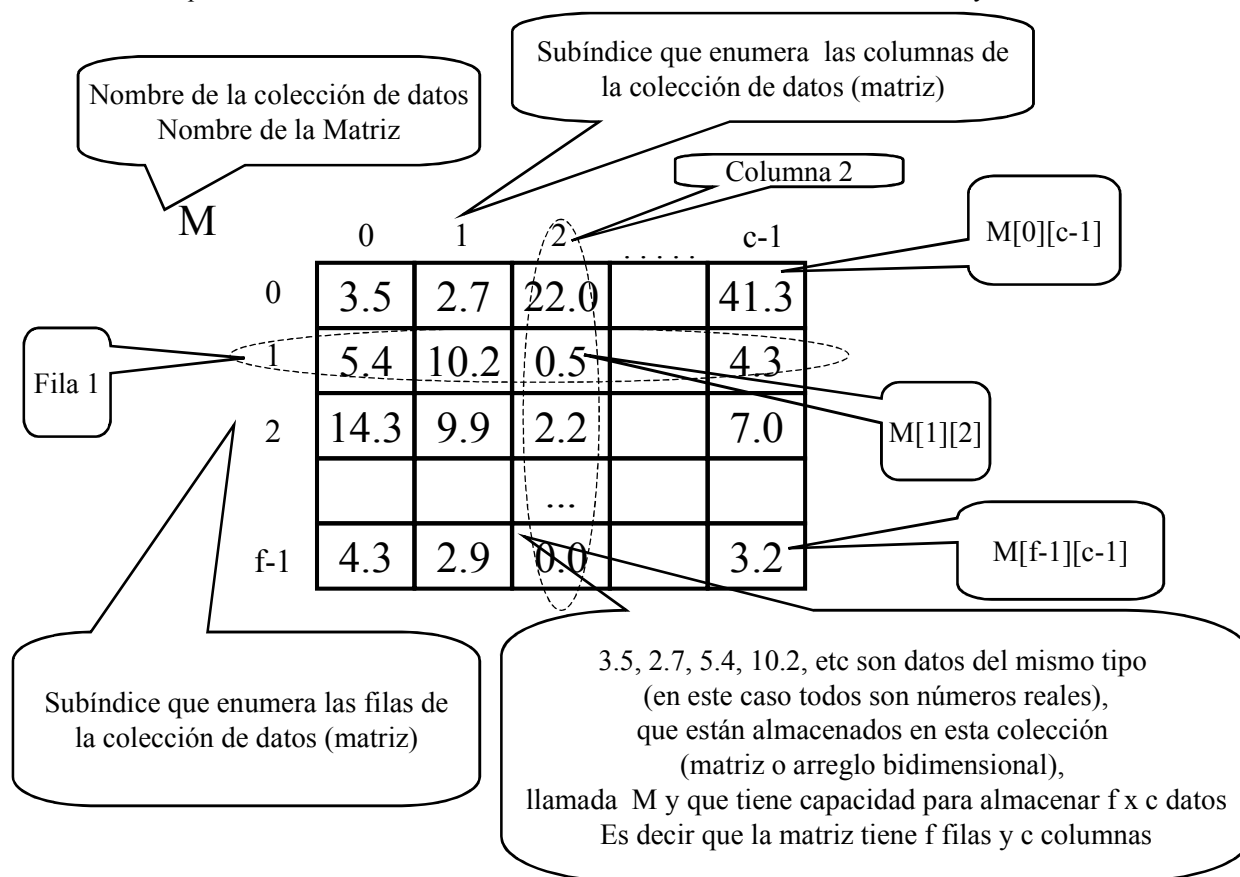
es 65), y se intercambia por el carácter correspondiente. Con la cadena encriptada debe ir un vector de enteros, que indica las posiciones donde hay espacios en blanco en la cadena original. El tamaño del vector debe ser igual al número de espacios en blanco. Se deben implementar las siguientes funciones:

- Capturar por teclado la cadena
- Llenar el vector con las posiciones de los espacios en blanco
- Eliminar espacios en blanco
- Pasar a mayúsculas
- Encriptar (reemplaza cada carácter como se ha indicado).
- Imprimir Código encriptado.

6 Arreglos Bidimensionales o Matrices

Una Matriz o arreglo bidimensional es una colección de datos del mismo tipo, referenciada con un solo nombre y que para acceder a cada uno de sus datos se necesita dos subíndices que indican la posición (la fila y la columna) donde se encuentra almacenado un dato, dentro de la colección.

En este caso los datos están organizados en dos dimensiones (Filas y columnas) y por esta razón se requiere de dos subíndices para ubicar un dato en particular. Uno de los subíndices indicará la fila donde se encuentra el dato y el otro subíndice la columna.



56- Capturar por teclado los datos de una matriz de 3x6 y posteriormente, sumar todos sus valores, determinar la cantidad de valores positivos y negativos dentro de la matriz, hallar la suma de cada columna y registrar en la fila 4 dichos valores, hallar el menor valor dentro de la matriz y la posición que ocupa.



57-Dada una matriz de 5x5 hacer un programa para llenarla y posteriormente, hallar la traza (la suma de la diagonal principal), hallar la suma de la diagonal secundaria, determinar si es un cuadrado mágico. Para ser un cuadrado mágico debe cumplir las siguientes reglas: la suma de la diagonal principal, secundaria, de las filas, columnas deben dar en todas lo mismo.

58-Dadas dos matrices de 5x8 hacer un programa para llenarlas y posteriormente, construir una tercera matriz con la suma de las dos anteriores.

59-Dada una matriz de 4x4 hacer un programa para llenarla y posteriormente, determinar si es una matriz identidad. Para ser matriz identidad debe cumplir con las siguientes reglas: Todos los elementos de la diagonal principal son 1 y los demás deben ser 0

60-Dadas dos matrices una de 5x3 y otra de 3x2 hacer un programa para llenarlas y posteriormente, crear una tercera matriz con la multiplicación de ambas.

61-Dada una matriz cuadrada de nxn hacer un programa para llenarla, una función que intercambiar los contenidos de dos números de fila dados como parámetros.

62-Elabore un programa que lea una matriz de m x n y genere una segunda matriz con las columnas como renglones(filas) y los renglones(filas) como columnas.

Por ejemplo, si la matriz que da el usuario es:

4	7	1	3	5
2	0	6	9	7
3	1	2	6	4

Entonces el programa debe escribir la matriz transpuesta:

4	2	3
7	0	1
1	6	2
3	9	6
5	7	4

63-(*) Elabore un programa que lea una matriz de 3 X 3 y calcule la inversa usando el método de Gauss-Jordan.

64- (*) Resuelva el siguiente sistema de ecuaciones lineales utilizando matrices para su representación y solución.

$$2x - 2y + z = 1$$

$$-x + y + z = 0$$

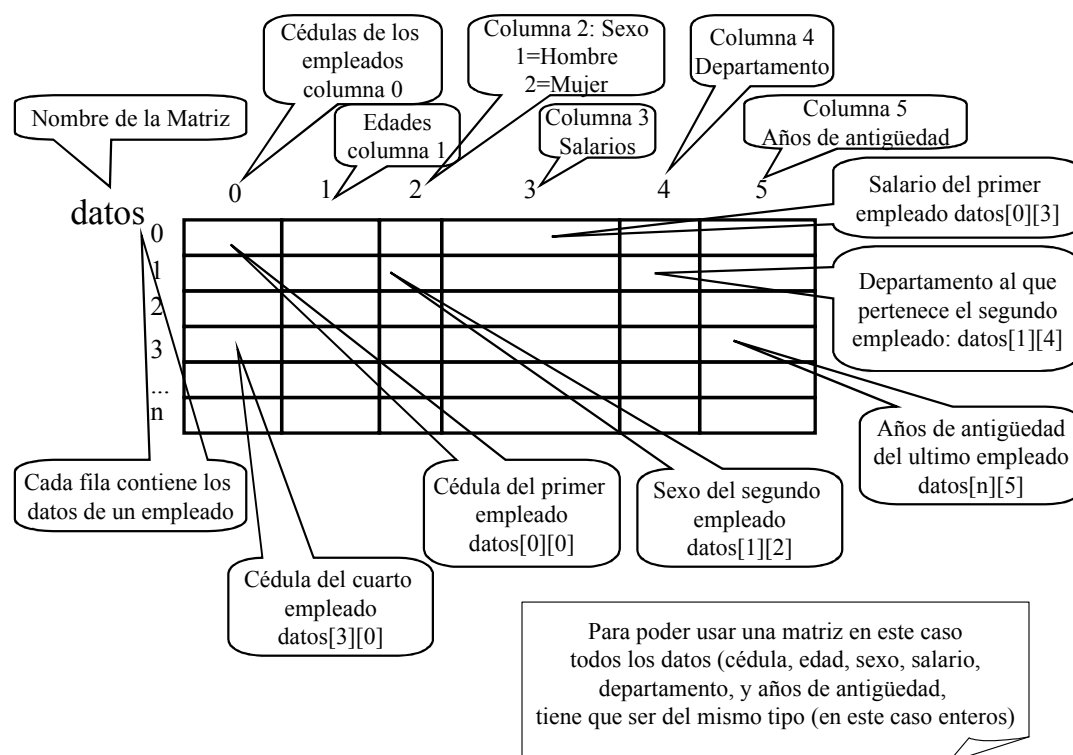
$$-x + 3y + 5z = 0$$

65-Elabore un programa que genere las tablas de multiplicar del 1 al 10 en un solo arreglo de dos dimensiones y posteriormente imprima dicho arreglo.

66-Una matriz cuadrada A se dice que es simétrica si $A(i, j) = A(j, i)$ para todo i, j dentro de los límites de la matriz. Elabore un programa que lea una matriz y diga si es simétrica o no

67-Una empresa tiene n empleados que trabajan en tres departamentos distintos (1001, 1002,1003) se ha elaborado un registro de cada uno de ellos: cedula, edad, sexo, salario, departamento y años de antigüedad. Elabore un programa que permita determinar:

- Número de empleados hombres y mujeres.
- Cedula de la persona con el salario mas bajo.
- Número de personas por departamento
- Cedula de la persona más antigua de la empresa.
- Cedula de la persona más joven de la empresa.



68-Hacer un programa que procese las notas de un grupo de estudiantes. Debe empezar solicitando la cantidad de alumnos y de notas a ingresar, almacenar el nombre del estudiante y las notas que obtuvo, tenga en cuenta que debe utilizar 2 matrices, una tipo char para los nombres y otra tipo float para las notas.

- Determine nombre del estudiante con el promedio más alto
- Determine nombre del estudiante con el promedio más bajo

69-Se tienen los resultados de las últimas elecciones a gobernador en el departamento X, el cual está conformado por 5 municipios. En dichas elecciones participaron 4 candidatos. (5X4). Elabore un programa que:

- Lea e imprima una tabla indicando los votos obtenidos en cada municipio por los 4 candidatos.
- Calcule el total de votos recibidos por cada candidato y el porcentaje del total de votos emitidos.
- Calcule el candidato más votado.
- Si un candidato recibió más del 50% de los votos, indicar que es el ganador. Si ningún candidato recibió más del 50% de los votos, el programa debe imprimir los dos candidatos más votados, que serán los que pasen a la segunda ronda de las elecciones.

70-En unas elecciones presidenciales del país, la registraduría va a sistematizar el conteo de votos; se leen los siguientes datos: número de mesa, votos por el candidato 54, 78 y 98, votos en blanco y nulos. El total de mesas fue de 200.345 en todo el país. Determinar:

- La cantidad de votos de cada candidato
- El número total de votos nulos
- El número total de votos en blanco
- El gran total de la votación
- El porcentaje de abstención si el potencial de electores es de 8000000 de habitantes.

71-Se desea registrar en una matriz la votación para presidente (# de votos). El país está compuesto por 11 departamentos. Se tienen 18 candidatos (representados en las columnas). Guardar en un vector el porcentaje de votación de cada candidato. Determine:

- El (los) candidato(s) que obtuvo (obtuvieron) mayor votación
- El candidato con menos votos expresado en porcentaje
- Los candidatos que fueron penalizados (votación menor o igual al 5%).



72- En el prestigioso colegio “El arca de Noe” estudian 1935 alumnos. Allí solo cursan 5 asignaturas y solo hay tres cursos 6,7 y 8 de cada uno se ha registrado la siguiente información correspondiente al segundo bimestre: Código del estudiante, curso, nota asignatura 1, nota asignatura 2, nota asignatura 3, nota asignatura 4, nota asignatura 5. Usted debe hacer un programa que permita establecer:

- El promedio de cada estudiante.
- El promedio de cada curso
- El código del mejor estudiante del colegio
- El promedio general del colegio
- Imprimir en orden descendente los cursos de acuerdo al promedio que se obtuvo en cada uno de ellos

73-En una empresa se va procesar la información de los 1240 empleados de cada uno ellos se conocen:

Código, sexo, salario básico, departamento al cual pertenece.

Los departamentos son:

01 Contabilidad

02 Administración

03 Mercadeo

04 Sistemas

Se debe determinar:

- Cuantas mujeres y cuantos hombres hay en cada departamento
- Cuantos empleados ganan menos de 100000
- Cuantas mujeres ganan más de 500000
- Cual es el hombre y la mujer que ganan más.
- Cual es el hombre y la mujer que ganan menos.

74-En un banco de la ciudad hay 1458 clientes que poseen cuenta corriente. Los saldos de las cuentas se actualizan al final del día alimentando el computador con las transacciones de todo el día, para cada cuenta se permite máximo una transacción se lee:

Número de cuenta, Cedula del Cliente, Saldo Anterior, Valor de la transacción, Tipo de transacción (1 deposito 2 Cheque girado)

Se desea imprimir:

- El listado de cuentas con el saldo actualizado
- Cuantas cuentas y quienes son sus titulares, quedaron con saldo en rojo (Saldo negativo)
- Cual es el total de dinero de sus clientes que posee el banco al final del día.
- Cual es el cliente que tiene más dinero en su cuenta y de cuanto es el monto
- Cuanto suman los cheques girados al final del día

75-Una empresa automotriz tiene cinco agencias y cuenta con la información acerca de las ventas mensuales de automóviles logradas el año pasado por cada una de éstas. A partir de estos datos la empresa construyó la siguiente matriz ventas:

	Lomas	Vallejo	Perisur	del Valle	Oriente
Enero					
Febrero					
...					
Noviembre					
Diciembre					

Elabore un programa para contestar las siguientes preguntas:

- ¿Cuál fue el total de ventas en el año de la agencia Lomas?
- ¿Cuál fue el promedio de ventas en el mes de diciembre?
- ¿Qué agencia tuvo mayores ventas en el mes de mayo?
- ¿En qué mes se registraron las menores ventas del año, considerando todas las agencias?

76-Un observatorio astronómico requiere de un programa que analice una fotografía del cielo tomada por la noche. La información de la fotografía está almacenada en forma de tabla, donde cada elemento representa la cantidad de luz que se registró para cada punto. Los valores registrados van del 0 al 20, por ejemplo:



0	3	4	0	0	0	6	8
5	13	6	0	0	0	2	3
2	6	2	7	3	0	10	0
0	0	4	15	4	1	8	0
0	0	7	12	6	9	10	4
5	0	6	10	6	4	8	0

La persona encargada de analizar la información supone que hay una estrella en (i, j) si: el punto no se encuentra en las orillas de la fotografía (primero o último renglón o columna), y $(a[i, j] + a[i - 1, j] + a[i + 1, j] + a[i, j - 1] + a[i, j + 1]) > 30$

Se espera como resultado del análisis, una tabla b con un "*" en las parejas (i, j) en las que se supone que hay una estrella. El resto de la tabla debe quedar lleno de espacios. La tabla b que resulta del ejemplo anterior es:

	1	2	3	4	5	6	7	8
1								
2		*						
3								
4				*				
5				*	*		*	
6								

Elabore un programa que:

- Lea las dimensiones de la tabla m y n con $(1 \leq m, n \leq 20)$.
- Lea los valores de cada elemento de la tabla a .
- Construya la tabla b .
- Imprima la tabla b .

77-Una fábrica de bombas hidráulicas tiene una matriz con los insumos necesarios para la producción de un conjunto de motores. Por ejemplo, suponiendo que la planta produce 7 motores y se utilizan 8 insumos en diferentes cantidades para su producción, la matriz sería:

		Insumos							
		1	2	3	4	5	6	7	8
M	1	10	20	30	40	0	60	10	80
O	2	0	70	0	50	40	30	0	10
T	3	5	10	15	0	10	15	5	0
O	4	10	20	10	20	10	0	10	20
R	5	4	0	8	0	6	8	4	0
E	6	0	6	9	12	15	0	1	24
S	7	20	18	0	14	0	10	8	6

Analizando la matriz anterior, podríamos ver que para producir un motor del tipo 3 es necesario utilizar 5 unidades del insumo 1, 10 unidades del insumo 2, 15 unidades del insumo 3, 10 unidades del insumo 5, 15 unidades del insumo 6 y 5 unidades del insumo 7. Por otro lado se tiene un arreglo con los costos unitarios en pesos de cada insumo:

Costo de Insumos

1	2	3	4	5	6	7	8
3.5	0.1	2.0	1.5	6.0	4.2	2.5	1.3

Se tiene otro arreglo con los pedidos a surtir en el presente mes:

Pedidos del mes de cada motor

1	2	3	4	5	6	7
100	25	75	150	80	90	10

También se tiene un último arreglo con la existencia actual de cada uno de los insumos:

Existencia de Insumos

1	2	3	4	5	6	7	8
120	0	20	60	40	90	10	0

Elabore un programa para:



- Capturar por teclado el número m de motores que fabrica la planta (máximo 50).
- Capturar por teclado el número n de insumos necesarios para la fabricación de los motores (máximo 50).
- Capturar por teclado los datos de la matriz de Insumos/Motor.
- Capturar por teclado los datos del arreglo de costos unitarios de insumos.
- Capturar por teclado los datos del arreglo de pedidos del mes de cada motor.
- Capturar por teclado los datos del arreglo de insumos existentes en la planta.
- Obtener e imprimir un arreglo de m elementos, con el costo de producción de cada motor. Suponga que el costo de producción de un motor consiste en sumar el producto de insumos necesarios por el costo unitario de cada insumo.
- Obtener e imprimir un arreglo de n elementos, con la cantidad de unidades de cada insumo, necesaria para cumplir con los pedidos del mes.
- Obtener e imprimir un arreglo de n elementos, cuyos datos indiquen cuál es el costo total por concepto de cada insumo para cumplir con los pedidos del mes.
- Obtenga e imprima el costo total (tomando en cuenta todos los insumos) para cumplir con la producción del mes.
- Obtenga e imprima un arreglo de n elementos, con la diferencia de los insumos necesarios para la producción mensual menos los insumos existentes en la planta, con el fin de poder surtir dichos insumos a tiempo.

78-En una empresa de taxis se desea registrar la información de las N (N dado por teclado) carreras realizadas por sus móviles (taxis) a los clientes de la empresa, existen 5 Taxis. Dada la siguiente información: Código del taxi, cédula del cliente, tipo de servicio y valor del servicio. Donde:

Código del Taxi
01
02
03
04
05

Tipo de servicio
01 Puerta a Puerta
02 Normal
03 Encomienda

Determine:

- Cual es el valor total producido por cada taxi.
- Cual es el servicio que más se usa.
- Que cliente paga más por un servicio.

Ejemplo de representación de la información en forma de matriz (arreglo bidimensional):

Número del taxi	Cédula Cliente	Tipo servicio	Valor
02	88030900	03	5000
03	60623890	01	2500
01	79120765	02	3000
.....

79- Se lee la información de 100 estudiantes, el código, la nota final, el departamento (1. SISTEMA, 2. ELECTRÓNICA, 3. TELECOMUNICACIONES), y el ciclo de formación al que pertenece (1 para los Primeros 5 semestres y 2 para los 5 últimos semestres). Representando esta información en una matriz. Se debe calcular e imprimir

- Cuantos estudiantes hay en cada departamento
- El código del estudiante con la nota más alta
- El promedio de notas de los estudiantes de primer ciclo, y segundo ciclo

80- Capturar por teclado los datos de una matriz de 3x4, las filas representan los productos 0,1,2 y las columnas representan las sucursales 0,1,2,3... en cada celda de la matriz se registra la cantidad mensual vendida de productos de la empresa. Se tiene además un vector de 3 posiciones con el costo de venta de cada producto.

Calcular e imprimir un vector de 4 posiciones con el valor total vendido por sucursal

Calcular e imprimir la sucursal que más dinero recaudo.

	0	1	2	3	4
0	1	2	3	5	1

Vector costo de productos

10000



1	2	1	2	2	2	15000
2	3	5	5	3	4	5000
Total Vendido	55000	65000	85000	95000	60000	

81-Un concesionario de automóviles alquila sus vehículos por días, dependiendo del tipo de vehículo los precios son así:

Tipo 01 Familiar valor \$20.000 diarios

Tipo 02 Deportivo valor \$30.000 diarios

Tipo 03 Camioneta valor \$50.000 diarios

Realice un programa que almacene la siguiente información Cédula, Tipo de vehículo, Días, Total a pagar, para N clientes (N es un número desconocido) El llenado de la información termina cuando la cédula del cliente es 0.

A los clientes con más de 8 días de alquiler en cualquier tipo se les hace un descuento del 5% sobre el valor total.

Realice con Funciones:

Función de Llenado (Solo puede solicitar cédula, Tipo de vehículo y días)

Función valor a pagar por cliente (almacenar este valor en la matriz)

Valor promedio de alquiler por tipo y general.

Cual es el tipo de vehículo que mas se alquila.

Ejemplo de Matriz:

Cédula	Tipo de Vehículo	Días	Valor a Pagar

82-La universidad de Pamplona desea llevar un registro de los 96 estudiantes deportistas vinculados a las selecciones de la universidad. Existen 3 selecciones

Código	Nombre
1015	Fútbol
1030	Rugby
1045	Taekwondo

De cada estudiante se conoce: Cédula, Sexo (1 Hombre, 2 Mujer), Edad, Código de la Selección a la que pertenece.

Realice con funciones: Llenado de la matriz. Determine el porcentaje de Hombres y mujeres (de toda la matriz). Determine la selección con el deportista más joven. Determine el número de deportistas por selección.

83-Una empresa de celulares quiere llevar el control de todos los planes que a vendido. Existen 528 clientes y de ellos se conoce: Cédula, Sexo (1 Hombre, 2 Mujer), Tipo de Plan (01 Familiar, 02 Estudiante y 03 Corporativo), Número de Celular y minutos consumidos. Los Planes son los siguientes:

Tipo de Plan	Minutos	Valor
01. Familiar	300	45.000
02. Estudiante	150	20.000
03. Corporativo	600	75.000

Realice:

- Main
- Función Llenado de la matriz que almacena los datos mencionados.
- Función que imprima el total que debe pagar cada cliente (debe tener en cuenta que al valor hay que sumarle los minutos extra si estos existen).
- Función que determine quien consume más minutos hombres o mujeres.
- Función cual es el cliente que más paga en su factura

Nota: Por cada minuto que se exceda un cliente en su plan debe pagar los minutos adicionales a 350 pesos cada uno.

84-Una empresa tiene 10 camiones tolva que trabajan los 7 días de la semana (Lunes a Domingo). Las horas trabajadas por día para cada camión se guardan en un arreglo BIDIMENSIONAL de 10 x7 elementos, con el fin de obtener después de cada semana la siguiente estadística:

- Los promedios de horas trabajadas cada día de la semana por los camiones.
- Los promedios de horas trabajadas por cada camión en la semana
- Deberá indicar el número de camión que trabajo un mayor número de horas durante la semana.
- En el main hacer el proceso de captura de información para el llenado del arreglo bidimensional (la matriz) de 10x7.

85-Realizar un programa en lenguaje C que permita el registro y control de láminas del álbum del mundial de fútbol en una matriz de 629 x 2N. Existen una cantidad N de aficionados. La información de cada aficionado se registra en dos columnas. Cada fila de la matriz representa una lámina. Por cada aficionado, en su primera columna se registra la existencia de la ficha: un 1 si la tiene y un 0 si no la tiene; en la siguiente columna se registra la cantidad de láminas repetidas que pueden ser intercambiadas (de tal manera que las columnas pares contienen la existencia de la lamina o no y las columnas pares contienen la cantidad de laminas repetidas que contiene cada aficionado) .

	Datos aficionado 1		Datos aficionado 2		Datos aficionado 3			Datos aficionado n	
	Existencia	Numero de láminas repetidas	Existencia	Numero de láminas repetidas	Existencia	Numero de láminas repetidas		Existencia	Numero de láminas repetidas
	0	1	2	3	4	5	...	2n-2	2n-1
0	1	5	0	0	1	14	...	1	0
1	0	0	1	5	0	0	...	1	2
...							...		
627	1	4	1	1	1	2	...	1	5
628	1	6	1	2	0	0	...	1	6

Desarrolle las siguientes funciones:

- Función de llenado de la matriz.
- Función que cuente la cantidad de láminas que le faltan para llenar el álbum a cada aficionado.
- Función que cuente la cantidad de fichas repetidas por cada aficionado.
- Función que indique el aficionado que tiene más fichas repetidas.
- Función main que llame las funciones e imprima los datos solicitados.

86-Una empresa dedicada al envío y entrega de giros de dinero, desea un programa en C usando matrices para almacenar la información de 32 giros enviados. De cada giro se almacena Código Ciudad Origen, Cedula de Quien Envía, Código Ciudad Destino, Cedula de quien Recibe y Valor del Giro. Los códigos de las ciudades son 1011, 1015 y 1020. Por ejemplo

	Código Ciudad Origen	Cc Envía	Código Ciudad Destino	Cc Recibe	Valor
	0	1	2	3	4
0	1020	60123456	1011	88321654	20000
1	1015	88030040	1020	88987654	50000



2
...
31					

Realice:

- Main
- Función Llenado de la Matriz
- Función que retorne el total enviado por Ciudad
- Función que retorne el promedio recibido por Ciudad
- Dado por teclado un número de Cedula Función que determine y retorne cuánto recibe esa persona por concepto de giros.

87-En una empresa de construcción se lleva el registro de las horas trabajadas por los obreros en cada uno de los proyectos. La información de las horas trabajadas se almacena en una matriz de $m \times n$ (m hace referencia al número de obreros, n hace referencia al número de proyectos.). La empresa cuenta con aproximadamente 1000 obreros y tiene como máximo 10 proyectos en ejecución. También se tiene la información del valor de la hora por cada proyecto. Ejemplo:

	Proyecto 0	Proyecto 1	Proyecto 2	...	Proyecto n-1
Obrero 0	25	80	0		10
Obrero 1	40	120	10		0
Obrero 2	50	125	1100		0
...			
Obrero m-1	40	56	40		60

Valor de la hora por proyecto	\$10.000	\$15.000	\$9.000	...	\$25.000
-------------------------------	----------	----------	---------	-----	----------

Hacer un programa en C que lea la información de la matriz de cantidades y el vector de valores y determine la siguiente información:

- Vector con la Cantidad total de horas trabajadas por cada uno de los obreros.
- Vector con el valor total pagado por mano de obra de cada proyecto.
- Indicar el proyecto más costoso de la empresa.
- Indicar la cantidad promedio de horas trabajadas por los obreros.

88- En una compañía de seguros se almacena la información de sus 150 afiliados en dos matrices de datos una en la que se almacenan el primer nombre y el primer apellido, y la segunda en la que se almacena el número de cedula, la edad, el sexo (1 Fem, 2 Masc), la antigüedad y el salario base de la persona.

Nombre y Apellido

	0	1	2	3	4	5	6	50
0	l	u	i	s		m	e	n	d	e	z	'\0'	
1													
...													
150													

	0 Cedula	1 Edad	2 Sexo	3 antigüedad	4 Salario Base
0					
1					
...					
150					

Realice un programa modular en lenguaje C en el que implemente lo siguiente:

- Función llenado: debe permitir Capturar por teclado la información de los 150 clientes almacenándola en las dos matrices de datos.

- Función imprimir nombre: debe imprimir el nombre de un cliente cualquiera (toma como referencia el número de la fila), imprimiendo el apellido en mayúsculas y el nombre en minúsculas separado por una coma (ej. MENDEZ, jorge) sin afectar la forma como se encuentra almacenado el nombre en la matriz de datos. Recuede que en el código ASCII A=65 Z=90, a=97 z=122.
- Función imprimir datos, debe imprimir los datos de una persona cualquiera (cédula, apellidos y nombres (función imprimir nombre), edad y sexo) teniendo como referencia el número de cédula.
- Función promedio edad: debe calcular y retornar el promedio de las edades de las personas registradas.
- Función personas en la edad promedio: debe imprimir en pantalla los datos de las personas que se encuentran en la edad promedio (usa las funciones promedioedad e imprimirdatos).
- Función rango salario: recibe dos salarios (salario inferior y salario superior) e imprimen en pantalla los datos de las personas que se encuentran en ese rango.
- La empresa igualmente tiene un plan de sorteos en el cual la persona ganadora recibe un premio de 2.000.000. Para determinar la persona ganadora se tiene en cuenta el día de la semana (1. Lun, 2. Mar, 3. Mier, 4. Jue, 5. Vier) y una consonante (b,c,d,f,g,....). El día de la semana corresponde a una vocal (1. a, 2. e, 3. i, 4. o, 5. u) la persona ganadora es aquella que tenga en su nombre y apellido más repetida la vocal (correspondiente al día de la semana) y la consonante ingresada. Realice una función que permita conocer el número de cédula de la persona ganadora (los datos de la vocal y consonante se reciben en minúscula pero debe contar así se encuentre en el nombre en mayúscula).
- Función vocal: recibe el número del día y retorna la vocal correspondiente en minúscula.

El main()

Realiza el llamado a la función llenado.

Realiza el llamado a la función personasen edad promedio.

Solicita dos salarios (inferior y superior) y realiza el llamado a la función rangosalario

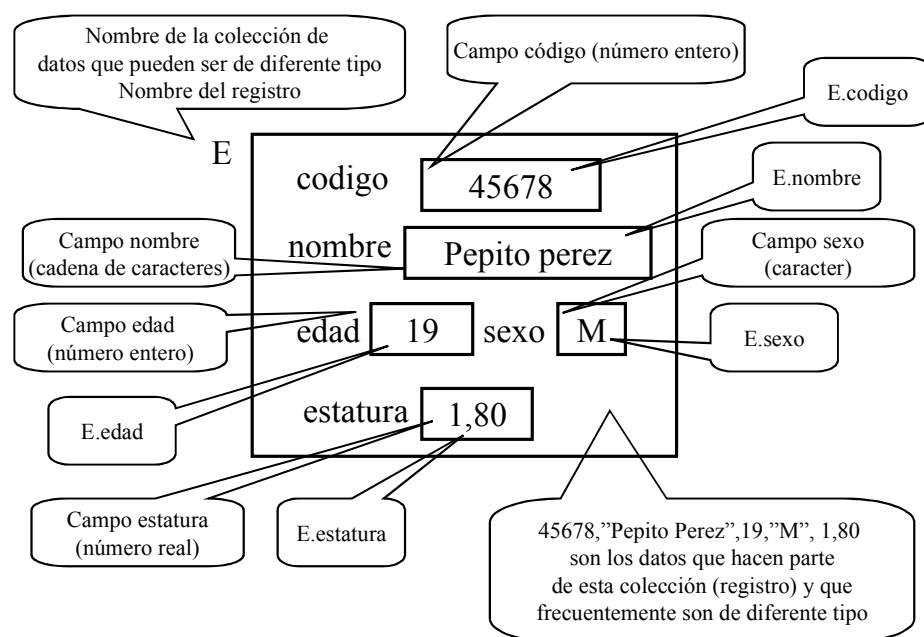
Solicita el día de la semana y una consonante y realiza el llamado a la función creada para conocer la persona ganadora.

Imprime los datos de la persona ganadora.

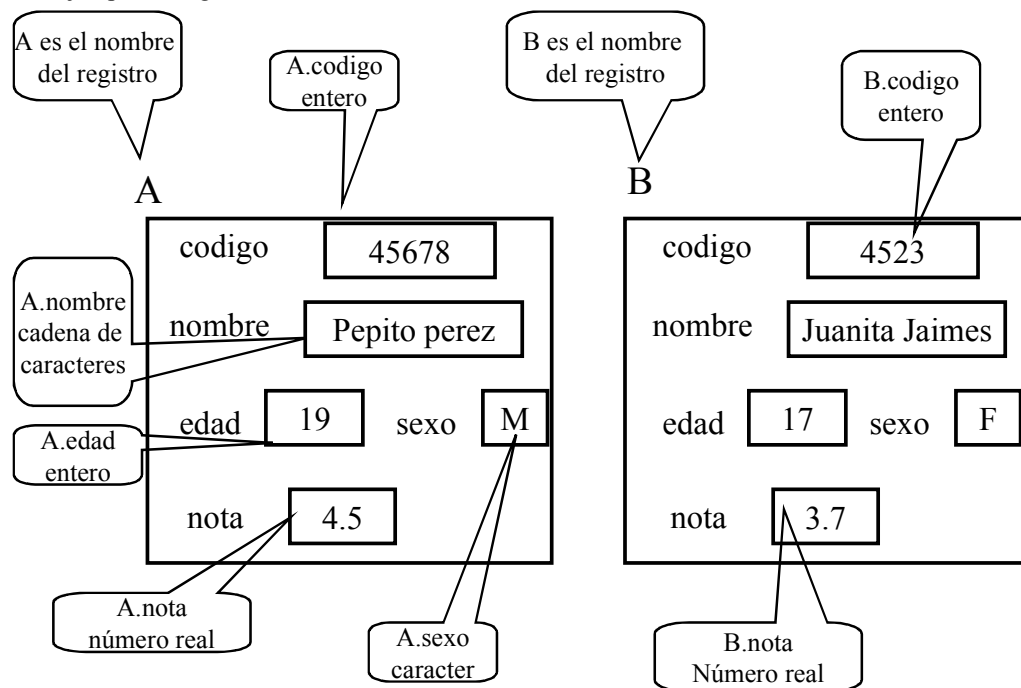
No se pueden usar las funciones gets y puts, ni el carácter de salida %s.

7 Registros

Los registros a diferencia de los arreglos son colecciones de datos que pueden albergar datos de diferentes tipos. A los datos que pertenecen a dicha colección se les denominan campos y se referencia con un nombre diferente para cada uno de los campos. Para tener acceso a cada uno de los datos de la colección (registro) se hace uso de la notación de punto así: `nombreregistro.nombrecampo`.

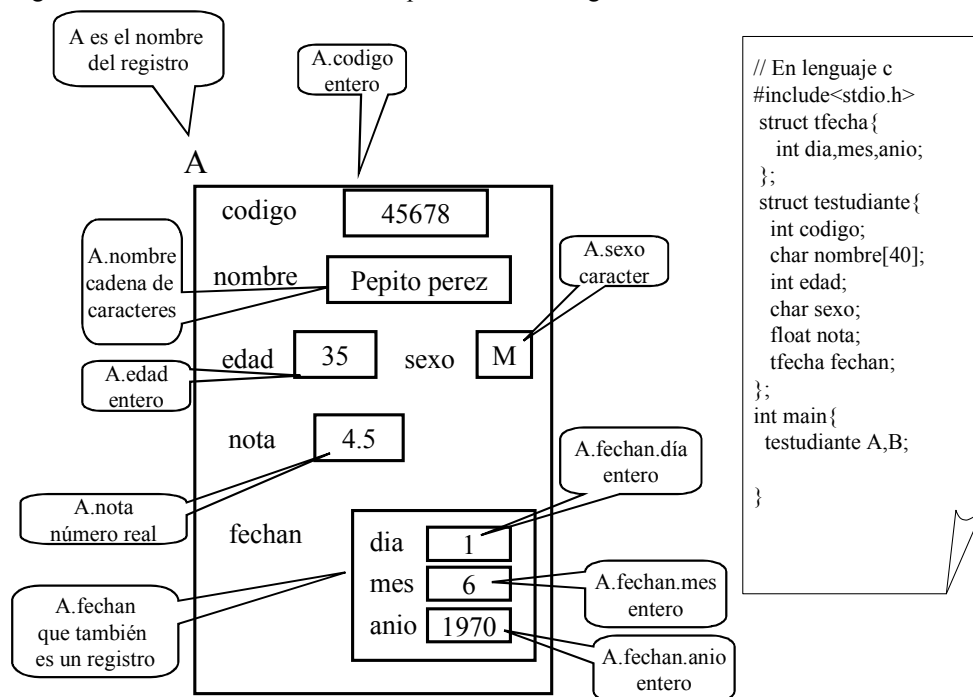


Otro ejemplo de registros



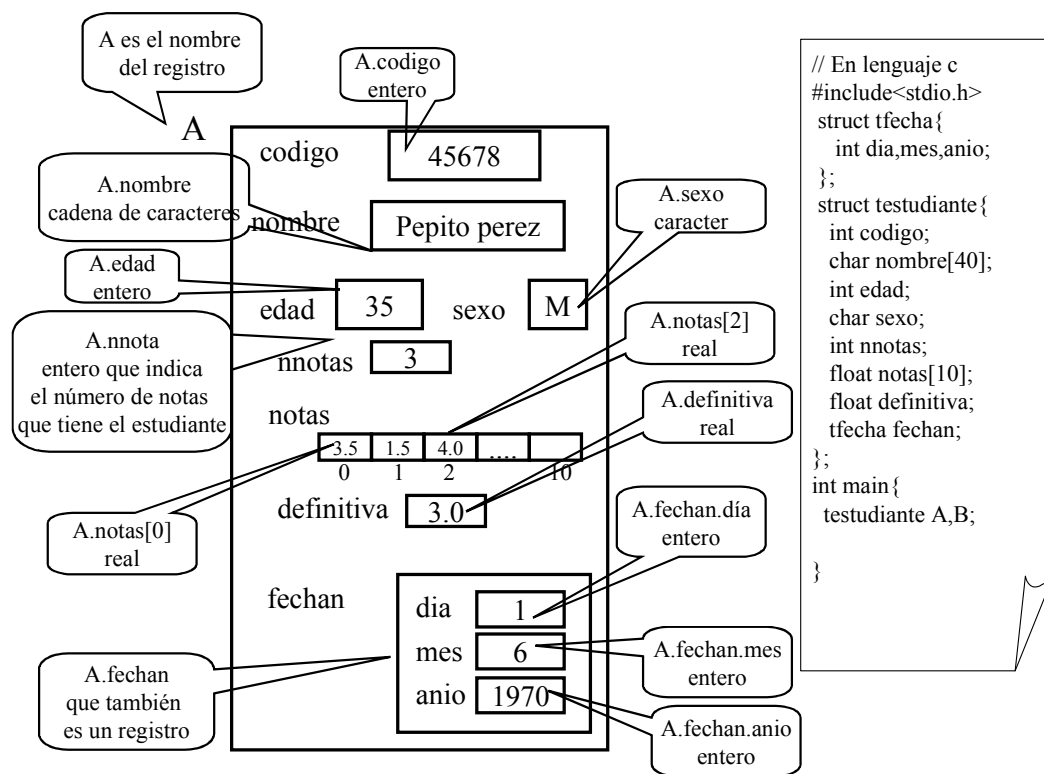
Registros Anidados

Registros en los cuales uno de sus campos también es registro



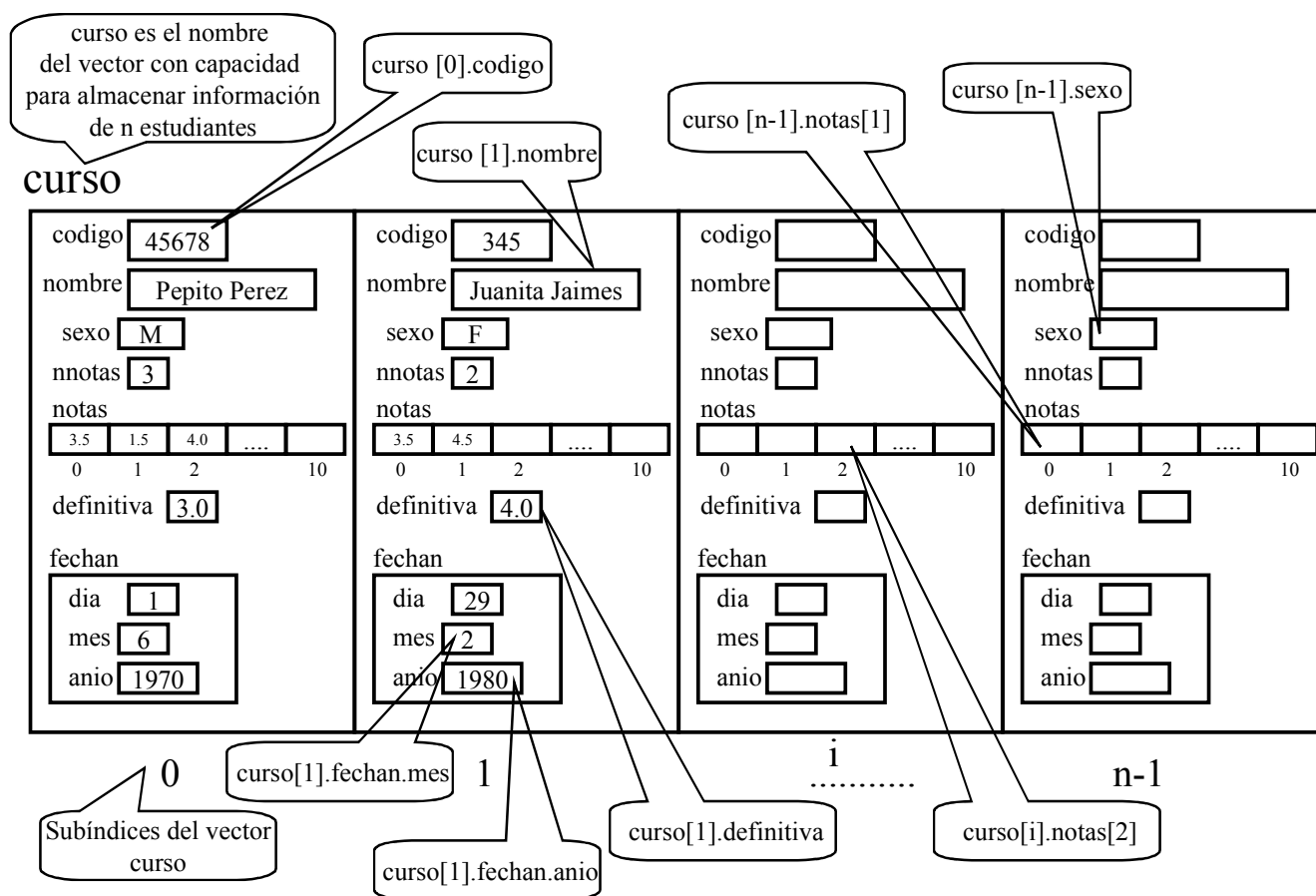
Registros con vectores

Ejemplo de registros en los cuales uno de sus campos es un vector



Vectores de registros

Un vector para el cual cada una de sus casillas almacena un registro



89-La información de todos los empleados de una cierta compañía está almacenada en una variable llamada “empresa” definida como un arreglo de registros. La información con que se cuenta de cada empleado es: nombre, sexo y sueldo. Escriba la declaración para el registro y el arreglo “empresa”.

Considerando las declaraciones que hizo, escriba las instrucciones en C para:

- Imprimir el nombre y el sexo del cuarto empleado.
- Imprimir los nombres de todos los empleados que ganan más de \$2,500.00
- Sumar al sueldo del tercer empleado \$2,000.00.

90-Se tiene un arreglo de una dimensión en el que cada uno de sus elementos tiene la siguiente estructura:

CLAVE	NOMBRE	SEMESTRE	MATERIAS
-------	--------	----------	----------

En **MATERIAS** se almacena información acerca de las 6 calificaciones obtenidas por un alumno en el semestre indicado por **SEMESTRE** (entero). Suponiendo que en el arreglo ya se almacena la información completa de n alumnos y que está ordenado en forma ascendente por clave, escriba:

- La definición del registro y la declaración de la variable donde se guarda la información de los n alumnos.
- Una función que busque en el arreglo una determinada clave, si ésta existe regrese como valor el lugar en el arreglo donde se encuentra. Si la clave buscada no existe deberá regresar un -1. Utilice búsqueda binaria.
- Un módulo que calcule el promedio de un cierto alumno.
- Un módulo que imprima la clave, nombre, semestre y promedio de un determinado alumno.
- Por último, escriba parte del programa principal donde se lean p claves y, utilizando los módulos anteriores, se escriban los datos de los p alumnos correspondientes.

91-En la Escuela Superior el valor de la matrícula de los 1800 estudiantes se determina según el número de materias que cursan. El costo de cada materia es \$ 400. Se ha establecido un programa para estimular a los estudiantes, el cual consiste en lo siguiente: si el



promedio obtenido por un estudiante en el último periodo es mayor o igual que 4.0, se le hará un descuento del 30% sobre el valor de la matrícula y no se le cobrará IVA; si el promedio obtenido es menor que 4.0 deberá pagar el valor de la matrícula completa, la cual incluye el 10% de IVA. Obtener: La estructura con la información: promedio, nombre, número_de_materias, capturada. El listado: promedio, nombre, número_de_materias y valor a pagar de los estudiantes con 30% de descuento. El valor recaudado por concepto de IVA

92-En el gran premio de Indianapolis, 100 participantes (cada uno) deben realizar 3 tiempos de clasificación. La información de cada corredor debe ser registrada en una estructura con los siguientes campos:

Código

Nombre

Tiempos [3] (vector de tipo estructura, donde cada posición tendrá los siguientes campos hora, minutos y segundos – Permite almacenar los 3 tiempos de clasificación)

Categoría (1. Novatos o 2. Profesional)

Sexo (1. Hombre o 2. Mujer)

Realizar: (implementando funciones)

- Capturar por teclado la información de cada participante en el vector de 100 posiciones de tipo estructura.
- Calcular e imprimir, el nombre del competidor con el menor tiempo en las carreras de clasificación en General.
- Calcular e imprimir, el nombre del competidor con el mayor tiempo en las carreras de clasificación en la categoría de los novatos.
- Calcular e imprimir, el nombre del competidor con el mayor tiempo en las carreras de clasificación en la categoría profesional.

93-Escriba un programa en Lenguaje C permita almacenar la siguiente información para un censo poblacional en un municipio:

Nombre de la persona (Cadena de caracteres)

Número de documento de identidad (Entero largo)

Tipo de documento ('C' = Cédula, 'T' = Tarjeta de Identidad)

Fecha de nacimiento (Compuesto por tres enteros uno para el día, otro para el mes y otro para el año)

Edad (Entero que debe guardar los AÑOS cumplidos de la persona)

El programa debe definir una estructura (registro) para almacenar la información de cada persona censada, para esto se utilizará un arreglo de registros del tipo definido anteriormente: Se sabe que el municipio tiene máximo 1000 habitantes.

El programa deberá realizar con funciones cada una de las siguientes tareas (una función para cada tarea):

- Capturar por teclado la información del censo para cada persona (Nombre, documento, tipo, fecha), EXCEPTO LA EDAD.
- Calcular la edad de las personas censadas, leyendo antes por teclado la fecha actual y retornar el promedio de edad de todos los censados.
- Ordenar la información de las personas ascendentemente de acuerdo a la edad y mostrar el listado ordenado.

Nota: Recuerde que el ordenamiento de un arreglo de registros se realiza de manera similar al de un arreglo normal.

94-Un grupo consta de n alumnos (máximo 50), donde cada alumno cursa 5 materias. La información con que se cuenta de cada alumno es el código (entero) y las 5 calificaciones (reales) de igual valor. Haga un programa que, utilizando un arreglo de registros:

- Lea la información de cada alumno y calcule la definitiva
- Calcule el promedio general de todos los estudiantes.
- Cuente cuantos estudiantes están por encima del promedio general

95-En una ciudad existe una empresa administradora de Restaurantes. Para llevar un mayor control de la información de sus ventas se desea un programa en C que haga uso de vectores de registros para procesar la información de las ventas que realiza cada uno de los restaurantes. Esta información estará almacenada en un vector como el que muestra la figura



<p>NitRestaurante <input type="text"/></p> <p>NombreAdministrador <input type="text"/></p> <p>NumMesas <input type="text"/></p> <p>Ventassemanales:</p> <table border="1"><tr><td><p>Fecha de venta</p><p>dia <input type="text"/></p><p>mes <input type="text"/></p><p>año <input type="text"/></p><p>ValorVenta <input type="text"/></p><p>NumPlatosvendi <input type="text"/></p></td><td><p>Fecha de venta</p><p>dia <input type="text"/></p><p>mes <input type="text"/></p><p>año <input type="text"/></p><p>ValorVenta <input type="text"/></p><p>NumPlatosvendi <input type="text"/></p></td><td></td><td></td></tr><tr><td>0</td><td>1</td><td>.....</td><td>6</td></tr></table> <p>TotalVentasde laSemana <input type="text"/></p>	<p>Fecha de venta</p> <p>dia <input type="text"/></p> <p>mes <input type="text"/></p> <p>año <input type="text"/></p> <p>ValorVenta <input type="text"/></p> <p>NumPlatosvendi <input type="text"/></p>	<p>Fecha de venta</p> <p>dia <input type="text"/></p> <p>mes <input type="text"/></p> <p>año <input type="text"/></p> <p>ValorVenta <input type="text"/></p> <p>NumPlatosvendi <input type="text"/></p>			0	1	6	<p>NitRestaurante <input type="text"/></p> <p>NombreAdministrador <input type="text"/></p> <p>NumMesas <input type="text"/></p> <p>Ventassemanales:</p> <table border="1"><tr><td><p>Fecha de venta</p><p>dia <input type="text"/></p><p>mes <input type="text"/></p><p>año <input type="text"/></p><p>ValorVenta <input type="text"/></p><p>NumPlatosvendi <input type="text"/></p></td><td><p>Fecha de venta</p><p>dia <input type="text"/></p><p>mes <input type="text"/></p><p>año <input type="text"/></p><p>ValorVenta <input type="text"/></p><p>NumPlatosvendi <input type="text"/></p></td><td></td><td></td></tr><tr><td>0</td><td>1</td><td>.....</td><td>6</td></tr></table> <p>TotalVentasde laSemana <input type="text"/></p>	<p>Fecha de venta</p> <p>dia <input type="text"/></p> <p>mes <input type="text"/></p> <p>año <input type="text"/></p> <p>ValorVenta <input type="text"/></p> <p>NumPlatosvendi <input type="text"/></p>	<p>Fecha de venta</p> <p>dia <input type="text"/></p> <p>mes <input type="text"/></p> <p>año <input type="text"/></p> <p>ValorVenta <input type="text"/></p> <p>NumPlatosvendi <input type="text"/></p>			0	1	6		
<p>Fecha de venta</p> <p>dia <input type="text"/></p> <p>mes <input type="text"/></p> <p>año <input type="text"/></p> <p>ValorVenta <input type="text"/></p> <p>NumPlatosvendi <input type="text"/></p>	<p>Fecha de venta</p> <p>dia <input type="text"/></p> <p>mes <input type="text"/></p> <p>año <input type="text"/></p> <p>ValorVenta <input type="text"/></p> <p>NumPlatosvendi <input type="text"/></p>																		
0	1	6																
<p>Fecha de venta</p> <p>dia <input type="text"/></p> <p>mes <input type="text"/></p> <p>año <input type="text"/></p> <p>ValorVenta <input type="text"/></p> <p>NumPlatosvendi <input type="text"/></p>	<p>Fecha de venta</p> <p>dia <input type="text"/></p> <p>mes <input type="text"/></p> <p>año <input type="text"/></p> <p>ValorVenta <input type="text"/></p> <p>NumPlatosvendi <input type="text"/></p>																		
0	1	6																
0	1	...	N - 1																

Hacer un programa en C que almacene la información de los n Restaurantes que administra la empresa (n dado por teclado) y las ventas que realizan en una semana cada uno de ellos. Realice las siguientes funciones:

- Una función que capture por teclado la información de los Restaurantes y sus Ventas
- Una función que dado un número de Nit retorne la totalidad de platos vendidos en la semana por dicho restaurante.
- Una función que calcule el valor total de las ventas de la semana de cada restaurante y almacene este valor en el campo TotalVentasde laSemana dentro de cada registro del vector de registros. Esta función debe retornar además el promedio total de ventas de todos los restaurantes que administra la empresa.
- Una función que retorne la posición (en el vector de registros) del restaurante que menos dinero recolectó por ventas durante la semana. Utilice esta función en el main para imprimir el nombre del administrador que menos dinero recolectó por ventas durante la semana
- Función main. En esta función se debe capturar el número de restaurantes que administra la empresa y hacer el llamado de la función que captura la información y la almacena en el vector de registros. Llamar a la función que retorna la totalidad de platos vendidos en la semana por un restaurante dado su Nit, la función que calcule y almacene el valor total de las ventas de cada restaurante e imprimir el promedio de ventas de la totalidad de restaurantes. Imprimir el nombre del administrador que menos dinero recolectó por concepto de ventas durante la semana.

96- Crear un programa que permite introducir cierta información relativa a los vuelos diarios que parte de un aeropuerto, en un Vector formado por registros (estructuras). Cada registro contendrá la siguiente información sobre el vuelo correspondiente:

Número de vuelo (no tiene que coincidir con el índice del arreglo)

Hora de partida (que a su vez tiene dos campos: Hora: 0...23 y Minuto: 0...59)

Origen del vuelo: Cadena de caracteres

Destino del Vuelo: Cadena de caracteres

Numero de pasajeros: Entero

Una vez introducido los datos de todos los vuelos se preguntará si desea obtener información de algún vuelo. En caso de que el usuario responda afirmativamente se pedirá el número de vuelo. El programa buscará el vuelo en el arreglo y accederá a la información que contiene a partir de su número de vuelo, mostrando por pantalla todos sus datos.

El programa se ejecutará repetitivamente hasta que el usuario indique que no desea obtener más información de ningún vuelo. Realizar funciones para:

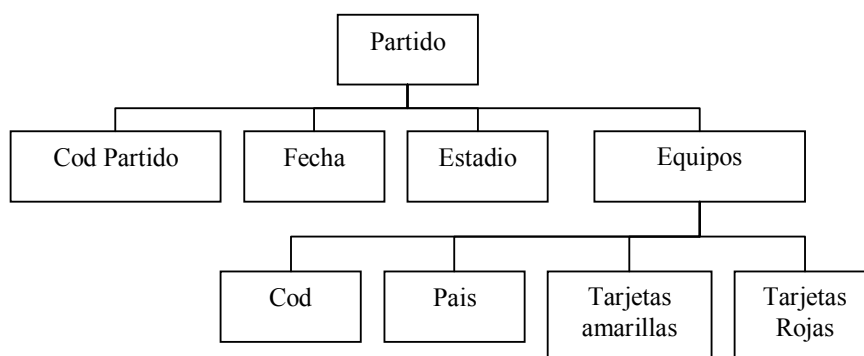
- Introducir la información de los vuelos en el arreglo
- Buscar un vuelo en el arreglo
- Visualizar un vuelo en pantalla

97- Los organizadores del mundial de fútbol le solicitan a un grupo de programadores al cual usted pertenece, un programa para almacenar la información de los 16 equipos que jugarán 32 partidos.

Utilice las siguientes estructuras:

Equipo: Código (entero largo), País (carácter 50 posiciones), goles (entero), tarjetas amarillas (entero), tarjetas rojas (entero).

Partido: Código Partido (entero largo), fecha (D/M/A enteras), estadio (carácter 50 posiciones), Equipos[2] (Vector 2 posiciones Estructura).



Cree las siguientes funciones:

- Función Llenar
- Función Multas Las tarjetas amarillas tienen un costo de 250 dólares y las rojas 500 dólares indique el total en multas por partido y por equipo en ese partido.
- Función Ganador Imprima el nombre del ganador de cada partido o empate.
- Función Goleador Imprima el nombre del equipo goleador del mundial.

98-Considere la siguiente definición y declaraciones

<pre>struct estudiante { char nombre[20], apellido [20]; flota calif[4];prom }; /* declaracion de la variables */ struct estudiante est1, est2;</pre>	<p>Indique para cada uno de las siguientes expresiones en C, si es correcta o no. En caso de ser incorrecta diga brevemente por qué.</p> <pre>printf("%s", est1); int i,s=0 for (i=0;i <4;i ++) s=s calif[i]; est2.prom = est1.prom; if (calif == calif) printf("Iguales"); if (est1 == est2) printf("Iguales"); if (est1.nombre == est2.nombre) printf("Iguales"); est2.prom=4</pre>
---	--

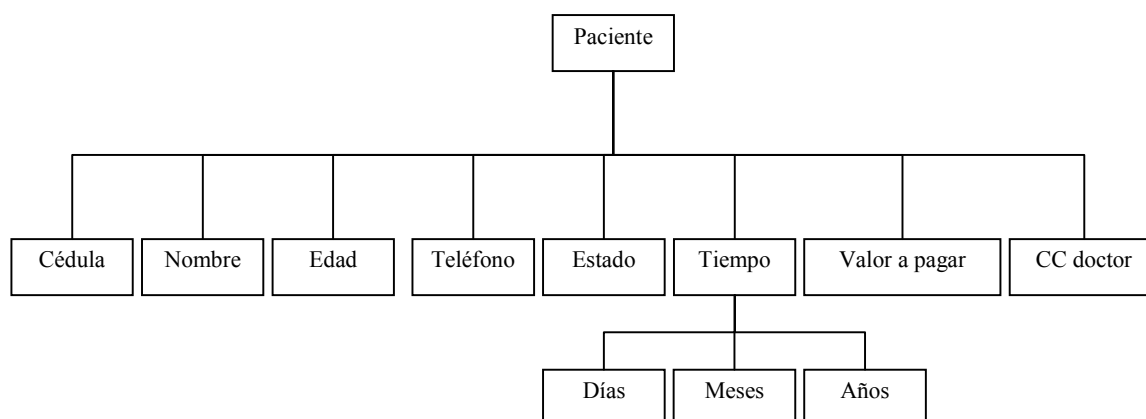
99-Realice un programa para controlar los datos de los pacientes de un hospital donde se manejan los datos básicos de los enfermos.

Pacientes: Cédula, Nombre, Edad, Teléfono, Estado del Paciente (1 Observación, 2 Cirugía, 3 En tratamiento), Tiempo (Estructura Interna), Valor a pagar, Cédula del Doctor Asignado.

Tiempo

Número de Días, Número de meses, Número de Años

** Tiempo es una estructura que almacenará el número de días, meses y años que lleva un paciente en el hospital (Asuma que los meses son de 30 días)



Realice:

- Llenar la información de los pacientes (el hospital puede aceptar máximo a 60 pacientes o hasta que la cédula del paciente sea 0).
- Indicar el nombre de los pacientes que se deben operar por un determinado doctor (la cédula del doctor dada por teclado).
- Determinar el valor a pagar de un paciente determinado (la cédula del paciente dada por teclado): Debe pasar los datos de la estructura Tiempo a Días y si el estado es Observación por cada día internado en el hospital debe pagar 4.000. Si el estado es Cirugía por cada día internado en el hospital debe pagar 6.000 y si el estado es Tratamiento por cada día internado en el hospital debe pagar 5.500.
- Determinar el porcentaje de pacientes en Observación, Cirugía y tratamiento
- El paciente de mayor edad que esté en tratamiento

100-Una factura típicamente contiene la siguiente información: No de factura, fecha, cliente, valor total, porcentaje IVA y un conjunto de ítems para los cuales se tiene un código, nombre de artículo, una cantidad, valor unitario.

Factura No: 8999898			
Nombre Cliente: Pepito perez			
Fecha día: 15 mes: 05 año: 2006			
Código	Nombre producto	Cantidad	Valor unitario
3458	Lapicero	10	800
3758	Marcador borrable	2	3.000
7896	Cuaderno de 100 hojas grapado	3	2.000
		Subtotal	20.000
		Iva: 16%	3.200
		Total	23.200

Dada las siguientes estructuras de datos en forma de registros:

<pre>struct tfecha { int dia, mes, anio; };</pre>	<pre>struct titem{ int codigo; char nombreprod[20]; int cantidad; float valorunitario; };</pre>	<pre>struct tfactura{ int no; char nombrecliente[40]; tfecha fecha; titem productos[20]; float subtotal; float total; };</pre>
---	---	--

Una papelería tiene almacenada la información de sus facturas (de un año) en un vector de registros de máximo 2000 posiciones
tfactura facturas[2000];

Elabore un programa en c con las siguientes funciones:



- Una función que capture los datos de facturas hasta que el número de factura sea cero. Como cada factura tiene un conjunto de ítems, para cada ítem capturar el código, nombre del producto, cantidad y valor unitario hasta que el código del producto sea cero. Con los datos de los ítems calcular el subtotal de la factura y el total a pagar, almacenándolos en los respectivos campos del registro factura.
- Hacer una función que calcule el total recaudado por la papelería en cada uno de los meses del año, para ello guardar la información en un vector de 12 posiciones.
- Hacer una función que calcule el total recaudado por concepto de IVA. (toda factura tiene un 16% de IVA)

101-De acuerdo a la siguiente estructura de datos (registro)

Struct Templeado{

Int codigo, horastrabajadas;

Char nombre[40];

Float salario, ventas, bonificacion;

};

Defina una variable en el main "empleados" que es un arreglo de registros de tipo "Templeado".

Construya un programa en C, que capture mediante una función, los datos básicos (codigo, nombre, horastrabajadas y ventas) de N empleados y liquide el salario de cada empleado de acuerdo a las siguientes condiciones: Cada hora se le paga a \$5000 pesos. Si las ventas son menores de 100000 pesos, se da una bonificación del 5% sobre las ventas. Si las ventas están entre 100000 y 500000 pesos, se da una bonificación del 8% sobre las ventas. Si las ventas son mayores de 500000 pesos se da una bonificación del 10% sobre las ventas

Hallar el salario promedio mediante una función que retorne dicho valor.

Mediante una función, Imprimir en pantalla los datos de todos los empleados

102-La empresa de buses Berlinas desea llevar el control de sus pasajeros por la temporada navideña. De ellos se conoce: Cédula, Nombre, edad, destino, peso equipaje. El valor del tiquete esta sujeto al destino y al peso del equipaje según la siguiente tabla

Destino	Valores
A	65.000
B	72.000
C	85.000

Todos los pasajeros tienen derecho a llevar 15 Kg de equipaje y por cada kilo adicional se incrementa el 2% del valor del tiquete por cada kilo adicional. Realice:

- Main
- Llenado.
- Cantidad de pasajeros por destino.
- El promedio de equipaje por pasajero en cada ruta.
- El cliente que más paga por exceso de equipaje.

Nota el llenado de la estructura es hasta que la cédula del pasajero sea igual a 0

** solo se permiten máximo 40 pasajeros por ruta

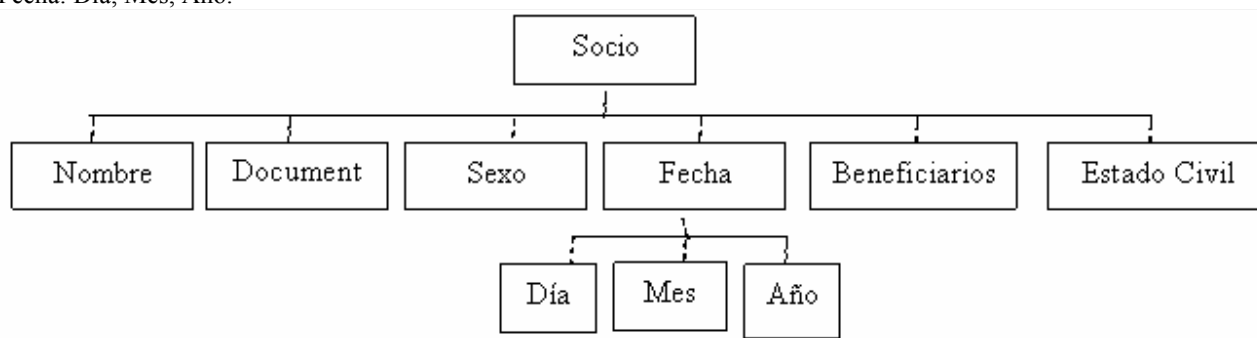
103-En una agencia de turismo, las N personas que van a pagar el importe de su tiquete aéreo o terrestre, llegan a la caja y sacan una bolita de color, que les dirá el descuento que tendrán sobre el total de su pasaje. Se sabe que si el color de la bolita es roja el cliente obtendrá un 40% de descuento; si es amarilla un 25% y si es blanca no obtendrá descuento. La captura de información termina cuando el valor del tiquete terrestre o aéreo sea 0. Se desea saber usando registros:

- El tiquete con la información básica del cliente y con el valor a pagar.
- La cantidad que pagaron los clientes que viajan por tierra cada día.
- El valor de los descuentos generados por la bolita roja en el día.
- El número de personas que no obtuvieron descuento en el día
- El porcentaje de clientes con tiquete aéreo que sacaron la bolita amarilla

104- Un centro de recreación quiere sistematizar la información de sus 534 socios de cada uno de ellos se conoce la siguiente información:

Socio: Nombre, Documento de Identidad, Sexo (Hombre (1), Mujer (2)), Fecha de Ingreso (Estructura Interna), Numero de Beneficiarios, Estado Civil (Soltero (0), casado (1)).

Fecha: Día, Mes, Año.



Realice:

- Declaración de la Estructura.
- Main
- Función de Llenado.
- Una función que cuente y retorne la cantidad de Socios solteros con más de 5 beneficiarios e imprima el nombre de dichos socios.
- Porcentaje de socios Hombres y mujeres.
- Recaudo por concepto de mensualidad del total de los socios, dependiendo de la tabla. Todos los socios con más de 15 años de antigüedad tienen un descuento del 5% sobre el valor a pagar.(Debe preguntar la fecha actual solo una vez para calcular los años de antigüedad)

Tipo de socio	Número de Beneficiarios	Valor a Pagar
Soltero	1 a 3	55.000
Soltero	Mayor de 3	78.000
Casado	1 a 4	25.000
Casado	Mayor de 4	35.000

105-La biblioteca de la Universidad necesita complementar un programa en C con el uso de registros que le permita tener un control sobre los préstamos. Aunque el programa ya existe, se requiere añadirle unas cuantas funciones que Ud tiene que implementar. Para ello se describe a continuación como está organizada la información dentro del software existente: Se tiene un vector de registros con la información de los estudiantes: código, nombre, teléfono, semestre en el que se encuentra matriculado el estudiante, el número de libros que tiene prestados el estudiante y los datos de los libros que tiene prestado.... es decir un vector de registros de tipo libro.

Cada préstamo esta registrado con la información del libro que tiene un código de inventario, un titulo, un autor y una fecha de entrega (día, mes y año)

<p>El encabezado del programa existente es el siguiente</p> <pre>#include<stdio.h> #include<stdlib.h> #include<conio.h> struct tfecha{ int dia,mes,anio; }; struct tlibro{ int noinv; //número de inventario char titulo[100]; char autor[40]; tfecha fechaentrega; };</pre>	<p>En el cuerpo del programa ya se tiene muchas instrucciones, pero se resumen las siguientes, que se consideran importantes para el trabajo que ud como programador tiene que realizar</p> <p>No necesita hacer la captura de datos por teclado, puesto que la información ya está en la memoria del computador, en el vector de tipo testudiantes y la variable nest que ya tiene el valor correspondiente.</p> <pre>int main(){ testudiante est[1000]; // vector donde se guarda la información de los estudiantes existentes en la universidad int nest; // total de estudiantes que se registraron en la universidad int diaactual, mesactual,anioactual,lv,semestre,total,pos; printf("Por favor teclee la fecha actual \n"); printf(" Dia : "); scanf("%d", &diaactual); printf(" Mes : "); scanf("%d", &mesactual);</pre>
--	---



<pre>struct testudiante{ int codigo; char nombre[40]; int telefono; int sem; //semestre en que se encuentra matriculado el estudiante int np; //número de libros prestados tlibro prestamo[10]; //vector para guardar los datos de los libros que tiene prestados el estudiante };</pre>	<pre>printf(" Año: "); scanf("%d", &anioactual); lv = clv(diaactual,mesactual,anioactual,nest,est); printf("existen %d libros con la fecha de entrega vencida"); printf("por semestre los estudiantes realizan la siguiente cantidad de prestamos"); for (semestre=1;semestre<=10; semestre=semestre+1){ total=contarporsemestre(nest,est,semestre); printf("total de libros prestados por los estudiantes de %d semestre es %d",semestre,total); }; pos=buscar(nest, est); printf(" el estudiante con más libros prestados es %s con código, %d y estudia en el semestre %d", est[pos].nombre, est[pos].codigo, est[pos].sem); }</pre>
--	--

Hacer las siguientes funciones

- Una función que dadas dos fechas (día, mes y año) devuelva un -1 si la primera fecha es anterior que la segunda, un cero si las fechas son iguales y un 1 si la primera fecha es posterior a la segunda. El prototipo de función es:
- int cmpfecha (int diax, int mesx, int aniox, int diay, int mesy, int anioy)
- Utilice la anterior función para hacer otra función llamada "clv" (contar libros vencidos) que devuelva el total de libros, cuyo préstamo ya se venció, es decir cuya fecha de entrega ya paso. Para ello la función recibe la fecha actual, y los datos de los estudiantes. Tenga en cuenta la forma del llamado realizada en el main.
- Realice una función que devuelva la posición del vector donde se encuentra el estudiante que tiene más libros prestados. Tenga en cuenta la forma del llamado realizada en el main.
- Realice una función que dado el número de un semestre, devuelva el total de libros que tienen prestado los estudiantes de ese semestre. Tenga en cuenta la forma del llamado realizada en el main.

106-Dada la información de N estudiantes, donde N se da por teclado:

Código del estudiante

Nombre del estudiante

Sexo del alumno (1 = Masculino, 2 = Femenino)

Código de la carrera a la cual pertenece

Donde la carrera se da en la siguiente tabla:

1	Ingeniería Sistema
2	Ingeniería Mecánica
3	Ingeniería Electrónica

Cada estudiante cursa 5 materias, por cada materia se da la siguiente información:

Código de la materia

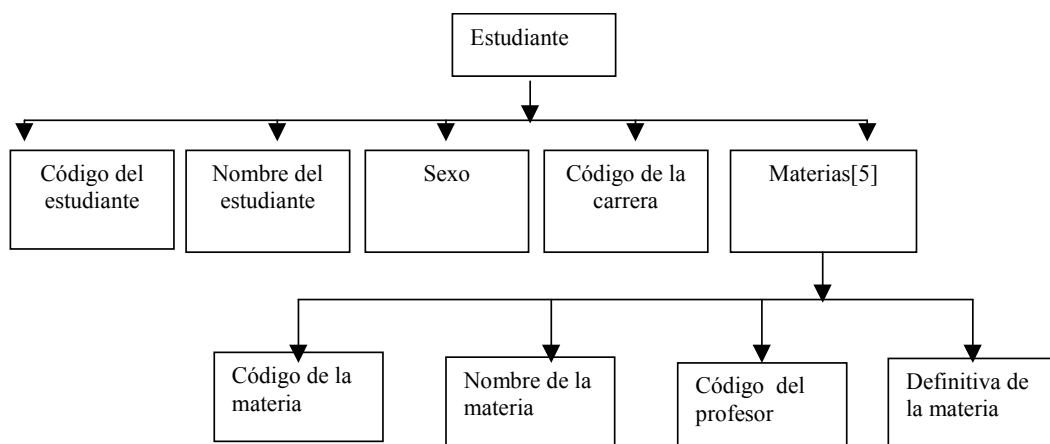
Nombre de la materia

Código del profesor de la materia

Definitiva de la materia

Codifique un programa en forma modular utilizando registros que realice lo siguiente:

- Una función que lea la información de cada estudiante y la almacena en registros
- Una función que calcule y retorne el código de la carrera que tiene mas cantidad de hombres
- Una función que calcule y retorne cual es el estudiante con mejor promedio
- Una función que calcule y retorne código de la carrera con mejor promedio



107-Una multinacional quiere que usted cree un programa para llevar el control de todos sus N clientes (N dado por teclado), de cada uno de ellos se conoce: Código del Cliente, Nombre, Cedula del cliente, tipo de cliente (1. Minorista, 2. Normal, 3. Mayorista) y número de productos. Cada uno de estos clientes tiene un número X de productos que maneja con la multinacional (X dado por teclado) de cada producto se maneja la siguiente información: Código del Producto, Nombre, Cantidad Pedida, Valor Unitario, Descuento. Cree:

- Main
- Función Llenado
- Función Cuanto debe pagar cada cliente por sus productos.
- Función Valor total pagado por cada tipo de cliente (1. Minorista, 2. Normal, 3. Mayorista).
- Función Cliente que más Paga por tipo de Cliente.

<p>Código <input type="text"/></p> <p>Nombre <input type="text"/></p> <p>Tipo <input type="text"/></p> <p>numpro <input type="text"/></p> <p>Productos</p> <table border="1"><tr><td>codigop <input type="text"/></td><td>codigop <input type="text"/></td><td>codigop <input type="text"/></td></tr><tr><td>nombrep <input type="text"/></td><td>nombrep <input type="text"/></td><td>nombrep <input type="text"/></td></tr><tr><td>cantidad <input type="text"/></td><td>cantidad <input type="text"/></td><td>cantidad <input type="text"/></td></tr><tr><td>valor <input type="text"/></td><td>valor <input type="text"/></td><td>valor <input type="text"/></td></tr><tr><td>des <input type="text"/></td><td>des <input type="text"/></td><td>des <input type="text"/></td></tr></table> <p>0 x-1</p>	codigop <input type="text"/>	codigop <input type="text"/>	codigop <input type="text"/>	nombrep <input type="text"/>	nombrep <input type="text"/>	nombrep <input type="text"/>	cantidad <input type="text"/>	cantidad <input type="text"/>	cantidad <input type="text"/>	valor <input type="text"/>	valor <input type="text"/>	valor <input type="text"/>	des <input type="text"/>	des <input type="text"/>	des <input type="text"/>	<p>Código <input type="text"/></p> <p>Nombre <input type="text"/></p> <p>Tipo <input type="text"/></p> <p>numpro <input type="text"/></p> <p>Productos</p> <table border="1"><tr><td>codigop <input type="text"/></td><td>codigop <input type="text"/></td><td>codigop <input type="text"/></td></tr><tr><td>nombrep <input type="text"/></td><td>nombrep <input type="text"/></td><td>nombrep <input type="text"/></td></tr><tr><td>cantidad <input type="text"/></td><td>cantidad <input type="text"/></td><td>cantidad <input type="text"/></td></tr><tr><td>valor <input type="text"/></td><td>valor <input type="text"/></td><td>valor <input type="text"/></td></tr><tr><td>des <input type="text"/></td><td>des <input type="text"/></td><td>des <input type="text"/></td></tr></table> <p>0 x-1</p>	codigop <input type="text"/>	codigop <input type="text"/>	codigop <input type="text"/>	nombrep <input type="text"/>	nombrep <input type="text"/>	nombrep <input type="text"/>	cantidad <input type="text"/>	cantidad <input type="text"/>	cantidad <input type="text"/>	valor <input type="text"/>	valor <input type="text"/>	valor <input type="text"/>	des <input type="text"/>	des <input type="text"/>	des <input type="text"/>
codigop <input type="text"/>	codigop <input type="text"/>	codigop <input type="text"/>																													
nombrep <input type="text"/>	nombrep <input type="text"/>	nombrep <input type="text"/>																													
cantidad <input type="text"/>	cantidad <input type="text"/>	cantidad <input type="text"/>																													
valor <input type="text"/>	valor <input type="text"/>	valor <input type="text"/>																													
des <input type="text"/>	des <input type="text"/>	des <input type="text"/>																													
codigop <input type="text"/>	codigop <input type="text"/>	codigop <input type="text"/>																													
nombrep <input type="text"/>	nombrep <input type="text"/>	nombrep <input type="text"/>																													
cantidad <input type="text"/>	cantidad <input type="text"/>	cantidad <input type="text"/>																													
valor <input type="text"/>	valor <input type="text"/>	valor <input type="text"/>																													
des <input type="text"/>	des <input type="text"/>	des <input type="text"/>																													

0

....

N-1

108-Para llevar los datos sobre los equipos participantes en el mundial y sus jugadores se requiere elaborar un programa en C, con una estructura de datos que soporte la siguiente información por cada equipo: Nombre del país participante, código FIFA, número de mundiales en que ha participado, número de goles recibidos, nombre del director técnico. Además por cada equipo se debe registrar el nombre de sus 23 jugadores, numero de pasaporte, la fecha de nacimiento, número de su camiseta, el número de mundiales en que ha participado, cantidad de goles anotados en este torneo. La estructura donde irá almacenada la información es similar a la del siguiente diagrama:



tequipo:		0	1	2	32
nomPais:	Brasil	codFIFA:	34			
numMun:	19	dirTecnico:	Carlos Caetano			
golContra:	3					
Jugadores:	22					
0.....		1	...	21		
nomJugador:	Julio Cesar					
pasaporte:	123456					
numCamiseta:	1					
fechaNaci:						
dia:	3	mes:	9	anio:	9	
numMun:	2	cantGoles	0			

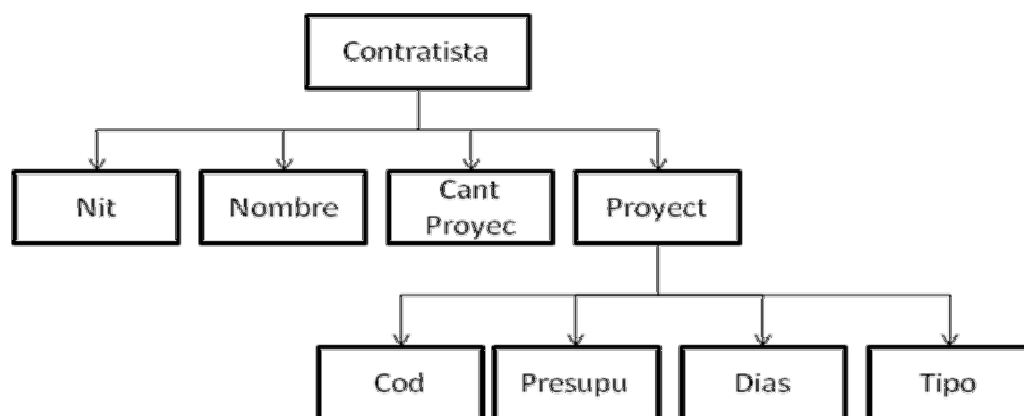
Implementar las siguientes funciones.

- Función llenado de la información
- Función que retorne el código FIFA del país con mayor cantidad de goles anotados. (suponemos que no hay empates).
- Función que retorne el código FIFA de la con la valla menos vencida del torneo (bajo el supuesto anterior).
- Función que retorne el promedio de goles del torneo.
- Función que retorne el pasaporte del goleador más joven del torneo (Esto es el jugador más joven que al menos ha metido un gol).
- Función main.

109- Se quiere controlar la información del Instituto de Desarrollo Urbano (IDU), donde existen N Contratistas (N dado por teclado). Los proyectos (obras) son asignados a un único contratista que será el encargado de desarrollar la obra. Un contratista puede tener adjudicado máximo 5 proyectos.

Los contratistas tienen un: Nit, Nombre y cantidad de proyectos.

Un proyecto tiene un código, un presupuesto, # de días en realizo el proyecto y un tipo de proyecto (1. Vias, 2. Reconstrucción y 3. Puentes). Todos estos datos son dados por teclado en el llenado inicial menos el #de días ya que este se llena cuando finaliza el proyecto.



Dependiendo del tipo de proyecto existe un plazo máximo de días para desarrollar la obra si se excede de estos límites debe cobrarse una multa que será proporcional al valor del día según el presupuesto por la cantidad de días de retraso.



Tipo de Proyecto	Días para su Ejecución
Vías	120
Reconstrucción	65
Puentes	345

Desarrolle un programa en C modular usando estructuras que llene la información de N contratistas y sus proyectos internos. Y posteriormente determine:

- Función llenar
- Función que reciba el código de un proyecto y los días que duro dicha obra y posteriormente almacene dentro de la variable días el valor correspondiente y calcule si el contratista paga o no una multa y a cuanto asciende esta.
- Función que reciba el Nit de un contratista y retorne la suma de todos los presupuestos de los proyectos que tiene a cargo.
- Función que determine que contratistas tienen a su cargo más proyectos de Reconstrucción.

110- En un parqueadero de la ciudad de Pamplona que atiende solo en horario diurno (en las noches el parqueadero queda desocupado), se cobra la tarifa por minutos de la siguiente manera, si el auto permanece cierto numero de horas mas 10 minutos no se le cobra los minutos excedidos de las horas enteras. Si la placa (numero de tres dígitos) del auto termina en número par, tiene un descuento de 20% sobre el total de valor. Todo auto paga un incremento del 16% de IVA sobre el valor a pagar. Si los dos últimos dígitos de la placa del vehículo coinciden con los dos últimos dígitos de la cedula del conductor, se realiza un descuento adicional del 25%.

La información sobre los conductores y los vehículos se debe almacenar en un vector de registros de la siguiente forma

Cientes

Cedula <input type="text"/>	Cedula <input type="text"/>	Cedula <input type="text"/>			
Nombre <input type="text"/>	Nombre <input type="text"/>	Nombre <input type="text"/>			
Placa <input type="text"/>	Placa <input type="text"/>	Placa <input type="text"/>			
Entrada	Entrada	Entrada			
Hora <input type="text"/>	Hora <input type="text"/>	Hora <input type="text"/>			
Minuto <input type="text"/>	Minuto <input type="text"/>	Minuto <input type="text"/>			
Salida	Salida	Salida			
Hora <input type="text"/>	Hora <input type="text"/>	Hora <input type="text"/>			
Minuto <input type="text"/>	Minuto <input type="text"/>	Minuto <input type="text"/>			
Valor <input type="text"/>	Valor <input type="text"/>	Valor <input type="text"/>			

0 1 2 499

111-Se requiere un programa en C que define las estructuras de datos necesarias para representar la información como se describe en el grafico.

Realizar las siguientes funciones:

- Una función que registre la entrada de un vehículo, capturando todos los datos del cliente (excepto la hora de salida y el valor a pagar) y almacenándolo en el vector en la posición correspondiente, según la llegada del cliente.
- Una función que dado un número de placa, retorne la posición dentro del vector donde se encuentra registrado dicho vehículo.
- Una función que registre la salida de un auto. Registre la hora y minutos de salida, Calcule el total a pagar (incluyendo descuentos e iva) por parte de un cliente dando dentro de los parámetros la posición donde se encuentra ubicado el registro respectivo, y el valor del minuto. Dentro de esta misma función se debe registrar este valor en el respectivo campo.
- Una función que calcule y retorne el total recaudado por el parqueadero en un día.
- En el main se debe, capturar el valor del minuto de parqueo, hacer uso adecuado de las funciones ya definidas y presentar un menú con las siguientes opciones:

1. Registro de entrada
2. Registro de salida
3. Terminar día



Este menú debe repetirse hasta que el administrador del parqueadero teclee la opción 3. En cuyo caso se debe imprimir en pantalla el total recaudado en el día.

112-Realizar un programa en C, que permita registrar la información asociada a los departamentos de Colombia y las instituciones educativas rurales y urbanas que existen en cada uno de los mismos. Por cada departamento se registra la siguiente información: Código, Nombre, Superficie (en Km2), Número de Habitantes

A su vez para cada departamento se tiene la información de las instituciones educativas con los siguientes datos:

Código, Nombre, Dirección, Número de estudiantes, Tipo (1. Urbana o 2 Rural)

Se desea que la información este representada en un vector de registros como se observa en la siguiente figura.

Codigo <input type="text"/>	Superficie <input type="text"/>	Codigo <input type="text"/>	Superficie <input type="text"/>	...	31
Nombre <input type="text"/>	nhabitantes <input type="text"/>	Nombre <input type="text"/>	nhabitantes <input type="text"/>		
NIstituciones <input type="text"/>		NIstituciones <input type="text"/>			
Instituciones					
Codigo <input type="text"/>	Codigo <input type="text"/>	Codigo <input type="text"/>	Codigo <input type="text"/>		
Nombre <input type="text"/>	Nombre <input type="text"/>	Nombre <input type="text"/>	Nombre <input type="text"/>		
Dirección <input type="text"/>	Dirección <input type="text"/>	Dirección <input type="text"/>	Dirección <input type="text"/>		
NEstudiantes <input type="text"/>	NEstudiantes <input type="text"/>	NEstudiantes <input type="text"/>	NEstudiantes <input type="text"/>		
Tipo <input type="text"/>	Tipo <input type="text"/>	Tipo <input type="text"/>	Tipo <input type="text"/>		
0	1	999	0	1	999

Realizar:

Definición de las estructuras necesarias para almacenar la información.

Una función que permita el Llenado de la información.

Función que retorne el departamento que tiene más estudiantes de tipo rural.

Función que retorne el total de estudiantes del país.

Función que retorne el departamento que tiene el mayor número de habitantes

Función Main. Realiza el llamado de las funciones e imprimir:El nombre del departamento que tiene más estudiantes de tipo rural, y la cantidad. La cantidad total de estudiantes del país. El nombre del departamento que tiene mayor número de habitantes, indicando estos datos.

NOTA:

El número máximo de instituciones educativas puede tomar un valor máximo de 1000 para algunos departamentos.

Recuerde que Colombia tiene 32 departamentos

113-En un concesionario, se reciben vehículos usados para la venta, hacer un programa en C que registre en un vector de registros la información de n vehículos que ingresan a un concesionario (N dado por teclado)

La información para cada vehículo es: Placa del vehículo, Marca(1. Mazda, 2. Ford), Color, modelo, Km recorridos, Fecha de vencimiento del seguro: (día, mes, año), Cantidad de dueños, Valor asignado

Para cada uno de los dueños (como parte del historial del vehículo) se debe saber: Nombre del dueño, Cédula, Teléfono, Valor de compra, Valor de venta

Elabore un programa que realice:

La definición de las estructuras de datos necesarias para guardar la información descrita.

Función que lea la información de cada vehículo

Función que devuelva la posición donde se encuentra el vehículo más costoso (valor asignado). Con esta información en el main se debe mostrar la placa de dicho vehículo, marca y color

Función que determina cuantos vehículos tienen fecha de seguro vencida con respecto a la fecha actual. En el main se debe pedir por teclado la fecha actual y entregarse dicha fecha a esta función como parámetro.

Función que determina para cada vehículo cual fue el dueño (nombre, cédula, teléfono) que más ganancia obtuvo.



Función que determina cual es el vehículo que ha recorrido menos cantidad de kilómetros, para los modelos 2000 a 2009.
Main que haga el respectivo llamado a las funciones e imprima los resultados.

8 Archivos

El tema de archivos en este curso tiene un alcance muy limitado ya que solo se aprovechan los archivos de texto como mecanismo para evitar la captura de datos por teclado, en el desarrollo de programas que incluyen arreglos de registros, donde la cantidad de datos a teclear es numerosa y por lo tanto se hace dispendiosa las pruebas de las funciones realizadas.

Aquí solo se utilizan los archivos de texto que contienen la información que normalmente se teclearía, y por lo tanto el programa no captura por teclado, sino que accede al archivo donde se encuentra la información, por lo tanto si se abre el archivo con un editor de texto se podría observar los datos en el mismo orden que se teclearían en caso de no usar los archivos.

El siguiente programa ilustra de manera general el uso de archivos para llenar la información de dos vectores de registros. El primer vector de registros contiene la información de estudiantes: código, nombre, sexo, fecha de nacimiento, número de materias matriculadas y los datos de las materias matriculadas que consta de dos campos: el código de la materia y la nota obtenida en dicha materia.

El segundo vector de registros contiene la información detallada de todas las materias existentes en un plan de estudios: código de la materia, nombre, intensidad horaria semanal, créditos y tipo de materia.

El main define dos vectores de registros llamados E y M de 100 posiciones cada uno, para guardar los datos de los estudiantes y de las materias respectivamente.

Si se realizara un programa que capture por teclado la información de 10 estudiantes y 10 materias como mínimo para hacer una prueba, sería necesario teclear por lo menos de 9 a 30 datos por cada estudiante y 6 datos por cada materia, esto implica mas o menos 160 datos dados por teclado cada vez que se ejecute el programa (esto podría llevar más de 3 minutos y se podrían presentar muchos errores de teclado), por lo tanto el uso de archivos de texto que contengan dicha información facilitaría la realización de pruebas con este volumen o mayor de datos, tan solo escribiendo una sola vez los datos en un archivo de texto (con cualquier editor de texto) y el programa en cuestión de milésimas de segundo carga la información en los vectores de registros diseñados para contener la información, por lo tanto no sería necesario teclear toda esta información cada vez que se ejecute el programa.

Gráficamente se puede representar las estructuras de datos (vectores de registros) como se muestra a continuación.

testudiante E[100]

<div>codigo <input type="text"/></div> <div>nombre <input type="text"/></div> <div>sexo <input type="text"/></div> <div>fechan <div>dia <input type="text"/></div><div>mes <input type="text"/></div><div>año <input type="text"/></div></div> <div>nmaterias <input type="text"/></div> <div>matricula<table><tr><td>codigom</td><td>codigom</td><td>...</td></tr><tr><td><input type="text"/></td><td><input type="text"/></td><td>...</td></tr><tr><td>nota</td><td>nota</td><td>...</td></tr><tr><td><input type="text"/></td><td><input type="text"/></td><td>...</td></tr></table></div> <div>012...9</div>	codigom	codigom	...	<input type="text"/>	<input type="text"/>	...	nota	nota	...	<input type="text"/>	<input type="text"/>	...	<div>codigo <input type="text"/></div> <div>nombre <input type="text"/></div> <div>sexo <input type="text"/></div> <div>fechan <div>dia <input type="text"/></div><div>mes <input type="text"/></div><div>año <input type="text"/></div></div> <div>nmaterias <input type="text"/></div> <div>matricula<table><tr><td>codigom</td><td>codigom</td><td>...</td></tr><tr><td><input type="text"/></td><td><input type="text"/></td><td>...</td></tr><tr><td>nota</td><td>nota</td><td>...</td></tr><tr><td><input type="text"/></td><td><input type="text"/></td><td>...</td></tr></table></div> <div>012...9</div>	codigom	codigom	...	<input type="text"/>	<input type="text"/>	...	nota	nota	...	<input type="text"/>	<input type="text"/>		
codigom	codigom	...																										
<input type="text"/>	<input type="text"/>	...																										
nota	nota	...																										
<input type="text"/>	<input type="text"/>	...																										
codigom	codigom	...																										
<input type="text"/>	<input type="text"/>	...																										
nota	nota	...																										
<input type="text"/>	<input type="text"/>	...																										
0	1	2	...	99																								

tmateria M[100]

<div>codigo <input type="text"/></div> <div>nombre <input type="text"/></div> <div>ihs <input type="text"/></div> <div>creditos <input type="text"/></div> <div>tipo <input type="text"/></div>	<div>codigo <input type="text"/></div> <div>nombre <input type="text"/></div> <div>ihs <input type="text"/></div> <div>creditos <input type="text"/></div> <div>tipo <input type="text"/></div>	<div>codigo <input type="text"/></div> <div>nombre <input type="text"/></div> <div>ihs <input type="text"/></div> <div>creditos <input type="text"/></div> <div>tipo <input type="text"/></div>			
0	1	2	3	...	99



<p>En el encabezado del programa se tienen las librerías necesaria para acceder a los archivos de texto y para la manipulación de cadenas de caracteres. De igual forma se definen 4 nuevos tipos de datos (en forma de registros) llamados tfecha, tnota, testudiante y tmateria. Estas estructuras sirven para la definición de los vectores de registros anteriormente representados en forma grafica.</p>		<p>Esta función captura una línea de texto “s”, carácter por carácter desde un archivo de texto referenciado por la variable “f”. Todos los archivos de texto se acceden línea por línea y la información es considerada como una cadena de caracteres, sin importar si el dato almacenado en la dicha línea sea un número. En una próxima función esta cadena capturada desde el archivo se convertirá al respectivo tipo de datos que corresponda (entero, real, char, etc)</p>
<pre>#include<stdio.h> #include<conio.h> #include<stdlib.h> #include<string.h> struct tfecha{ int dia,mes,anio; }; struct tnota{ int codigom; float nota; };</pre>	<pre>struct testudiante{ int codigo; char nombre[50]; char sexo; tfecha fechan; int nmaterias; tnota matricula[10]; }; struct tmateria{ int codigo,ihs,creditos,tipo; char nombre[20]; };</pre>	<pre>void capturarlinea(FILE *f,char s[]){ char c; int i=0; s[0]='\0'; c=fgetc(f); while((c!='\n')&&(!feof(f))){ s[i]=c; i=i+1; c=fgetc(f); }; s[i]='\0'; };</pre>
<p>Esta función accede al archivo de texto llamado “estud.txt”, el cual debe contener la información de estudiantes, que faciliten el llenado del vector de registros ya descrito E.</p> <p>La función hace uso de la función capturarlinea que equivaldría a un “gets”, utilizado para capturar por teclado... solo que esta función capturarlinea captura el dato desde un archivo de texto. Posterior a cada captura de datos, se realiza la respectiva conversión de datos, es decir que la cadena de caracteres es convertida al tipo correspondiente del campo dentro del respectivo registro. Obsérvese que la primera línea de entrada corresponden al numero de estudiantes que contiene el archivo de texto y las restantes líneas corresponden a los datos de cada estudiante, en el mismo orden que se hubiesen pedido por teclado</p> <pre>int leerdearchivoestudiantes(testudiante X[]){ FILE *entrada; char cadena[100]; int k=0,nx,i; if((entrada=fopen("estud.txt","rt"))==NULL){ printf("no se encontro el archivo estud.txt\n"); return (-1); }else{ capturarlinea(entrada,cadena); nx=atoi(cadena); printf("Se encontraron %d estudiantes en el archivo\n\n",nx); while (k<nx){</pre>		<p>Esta función accede al archivo de texto llamado “materia.txt”, el cual debe contener la información de materias, que faciliten el llenado del vector de registros ya descrito M.</p> <p>La función hace uso de la función capturarlinea. Posterior a cada captura de datos, se realiza la respectiva conversión de datos, es decir que la cadena de caracteres es convertida al tipo correspondiente del campo dentro del respectivo registro.</p> <p>Obsérvese que la primera línea de entrada corresponden al numero de materias que contiene el archivo de texto y las restantes líneas corresponden a los datos de cada materia, en el mismo orden que se hubiesen pedido por teclado.</p> <pre>int leerdearchivomaterias(tmateria X[]){ FILE *entrada; char cadena[100]; int k=0,nx,i; if((entrada=fopen("materia.txt","rt"))==NULL){ printf("no se encontro el archivo materia.txt\n"); return (-1); }else{ capturarlinea(entrada,cadena); nx=atoi(cadena);</pre>



<pre>capturarlinea(entrada,cadena); X[k].codigo=atoi(cadena); capturarlinea(entrada,cadena); strcpy(X[k].nombre,cadena); capturarlinea(entrada,cadena); X[k].sexo=cadena[0]; capturarlinea(entrada,cadena); X[k].fechan.anio=atoi(cadena); capturarlinea(entrada,cadena); X[k].fechan.mes=atoi(cadena); capturarlinea(entrada,cadena); X[k].fechan.dia=atoi(cadena); capturarlinea(entrada,cadena); X[k].nmaterias=atoi(cadena); for(i=0;i<X[k].nmaterias;i=i+1){ capturarlinea(entrada,cadena); X[k].matricula[i].codigom=atoi(cadena); capturarlinea(entrada,cadena); X[k].matricula[i].nota=atof(cadena); }; k=k+1; }; fclose(entrada); return k; }; };</pre>	<pre>printf("Se encontraron %d materias en el archivo\n\n",nx); while (k<nx){ capturarlinea(entrada,cadena); X[k].codigo=atoi(cadena); capturarlinea(entrada,cadena); strcpy(X[k].nombre,cadena); capturarlinea(entrada,cadena); X[k].ihs=atoi(cadena); capturarlinea(entrada,cadena); X[k].creditos=atoi(cadena); capturarlinea(entrada,cadena); X[k].tipo=atoi(cadena); k=k+1; }; fclose(entrada); return k; }; };</pre>
<p>Esta función imprime el vector de registros de estudiantes, que sirve para verificar si la información contenida en el archivo de texto fue adecuadamente almacenada en el vector de registros</p> <pre>void imprimirestudiantes(int nx,testudiante X[]){ int i,k; for(i=0;i<nx;i=i+1){ printf("\nCodigo: %d",X[i].codigo); printf("\nNombre: %s",X[i].nombre); printf("\nSexo: %c",X[i].sexo); printf("\nFecha de nacimiento: %d/%d/%d",X[i].fechan.anio,X[i].fechan.mes,X[i].fechan.dia); printf("\nNumero de materias matriculadas %",X[i].nmaterias); printf("\nNotas obtenidas:"); for(k=0;k<X[i].nmaterias;k=k+1){ printf("\n\t %d - >%2.1f",X[i].matricula[k].codigom,X[i].matricula[k].nota); }; printf("\npresione una tecla para mostrar el siguiente.."); getchar(); }; };</pre>	<p>Esta función imprime el vector de registros de materias, que sirve para verificar si la información contenida en el archivo de texto fue adecuadamente almacenada en el vector de registros.</p> <pre>void imprimirmaterias(int nx,tmateria X[]){ int i,k; for(i=0;i<nx;i=i+1){ printf("\nCodigo: %d",X[i].codigo); printf("\nNombre: %s",X[i].nombre); printf("\nlhs: %d",X[i].ihs); printf("\nCredito: %d",X[i].creditos); printf("\nTipo: %d",X[i].tipo); printf("\npresione una tecla para mostrar el siguiente..."); getchar(); }; };</pre>



El main define las variables E y M como vectores de registros de estudiantes y materias respectivamente, invoca a las funciones que realizan la captura de datos desde el archivo de texto respectivo y finalmente imprime los dos vectores de registros

```
int main(){
    testudiante E[100];
    int ne,nm;
    tmateria M[100];
    clrscr();
    ne=leerdearchivoestudiantes(E);
    imprimirestudiantes(ne,E);
    nm=leerdearchivomaterias(M);
    imprimirmaterias(nm,M);
    return 0;
};
```

Archivo estud.txt

Esta es la información que contiene el archivo de texto sobre los estudiantes.

Obsérvese que la primera línea corresponden al numero de estudiantes que contiene el archivo de texto y las restantes líneas corresponden a los datos de cada estudiante,

3
111
Pepito Perez
m
1982
3
15
3
10
4.2
11
1.5
12
3.2
222
Juanito Jaimes
m
2000
10
20
2
10

Archivo materia.txt

Esta es la información que contiene el archivo de texto sobre los estudiantes.

Obsérvese que la primera línea corresponden al numero de materias que contiene el archivo de texto y las restantes líneas corresponden a los datos de cada materia

4
10
Calculo Diferencial
4
4
1
11
Prog Estructurada
5
3
2
12
Lab de Mecanica
4
2
3
13
Mecanica
4
4
1



1.2 12 2.9 444 Pedrito polania f 2004 12 11 5 10 4.5 12 3.4 13 3.0 15 2.1 16 3.8	
---	--

Comprendido el código y la forma de utilización de los archivos de texto como mecanismo para capturar datos de entrada, se puede complementar el programa agregando más funciones y si es necesario agregando mas campos a las estructuras definidas:

114- Agregue al programa anterior las funciones necesarias para:

- Dado el código de un estudiante, calcular su promedio. Recordar que el promedio se calcula multiplicando cada una de las notas por el número de créditos de la materia correspondiente y al final dividir entre el total de créditos cursados.
- Dado el código de una materia calcular el promedio de dicho curso. Para ello hay que buscar todos los estudiantes que cursaron dicha materia y sumar sus notas, para al final dividir entre el total de estudiantes que la cursaron.
- Dado el código de una materia, imprimir la planilla de calificaciones, es decir el listado de estudiantes con su respectiva nota.
- Encontrar el mejor estudiante e imprimir todos sus datos registrados
- Encontrar la posición donde se encuentra el estudiante mas joven
- Hallar el mejor estudiante en programación que sea mujer.
- Cual es el estudiante que tiene mas créditos cursados
- Encontrar la materia donde más reprueban los estudiantes
- Calcular el promedio de notas de las materias teórico-prácticas
- Y muchas otras funciones que se pueden definir con la información registrada en los archivos de texto y con las estructuras de datos usadas.

115- Realice un programa que lea un archivo de texto codificado (codificado.txt) y que genere otro archivo con la decodificación del texto (decodificado.txt). La decodificación consiste en sustituir la primera 'w' de cada línea por 'a', la segunda 'w' por 'e', la tercera 'w' por 'i', la cuarta 'w' por 'o' y la quinta 'w' por 'u'. Si hubiera más de 5 'w' en una línea, deberá iniciarse nuevamente con la 'a'. El programa debe utilizar una función que decodifique una línea.

Ejemplo:

Si el archivo de entrada fuera

Lw cwsta de cwtricws hwele rico. Ponlw sobrw
lw mwsa

El archivo de salida deberá contener:

La cesta de cítricos huele rico. Ponla sobre
la mesa

116- En un archivo se tiene información sobre los miembros de un coro. La información que se tiene de cada uno es:

Nombre.

Parte de voz: soprano, alto, tenor o grave.

Extensión: alta o baja



Escriba:

- La definición del registro y la declaración de la variable para guardar la información de los miembros del coro.
- Un módulo para capturar la información del archivo y almacenarla en el vector.
- Un módulo que calcule el porcentaje de miembros del coro que tienen una determinada parte de voz.
- Un módulo que calcule el porcentaje de los miembros de una cierta parte de voz que tienen una determinada extensión. Por ejemplo, el porcentaje de los tenores que tienen extensión baja.
- Un módulo que dado un nombre escriba su parte de voz y extensión.
- Un módulo que calcule el total de miembros de cada una de las 4 partes de voz.
- La definición del registro y la declaración de la variable para guardar la información de los miembros del coro.
- Un módulo para capturar la información del archivo y almacenarla en el vector.
- Un módulo que calcule el porcentaje de miembros del coro que tienen una determinada parte de voz.
- Un módulo que calcule el porcentaje de los miembros de una cierta parte de voz que tienen una determinada extensión. Por ejemplo, el porcentaje de los tenores que tienen extensión baja.
- Un módulo que dado un nombre escriba su parte de voz y extensión.
- Un módulo que calcule el total de miembros de cada una de las 4 partes de voz.