

LENGUAJE C - PRACTICA

CONSTANTES, VARIABLES Y OPERADORES

- 1- Dada la sigte línea: **#define VALOR 15** marcar en cada caso la/s respuestas correctas:

Cuál o cuales de las sgtes. sentencias son válidas?

- a. `VALOR += i;`
- b. `printf ("%s", VALOR);`
- c. `++ VALOR;`
- d. `printf ("%d", VALOR);`
- e. ninguna de las anteriores

VALOR es:

- f. Constante
- g. Variable inicializada
- h. Constante de carácter
- i. Constante simbólica
- j. Ninguna de las anteriores

- 2- Definir la constante simbólica DOLAR dándole el valor 3.18.

.....

- 3- Habiendo definido la constante simbólica DOLAR, esta ¿puede cambiar su valor en tiempo de ejecución?

SI - NO

- 4- Decir si las siguientes expresiones son válidas:

a. **#define** PI 3.141592, NUMERO_E 2.718281

b. **#define** FIESTA "7 de julio \"San Fermín\""

- 5- Dada la siguiente directiva al preprocesador

#define NOMBRE "Raquel"
si escribimos:

`printf("Hola %s", NOMBRE);`
por pantalla se verá:

- 6- Decir cuales de los siguientes identificadores son válidos, si no lo son decir porque:

a) registro1	d) return	g) nom y dir	j) 123-45-6
b) lregistro	e) \$impuesto	h) nom_y_dir	k) nom*ap
c) archivo_3	f) nombre	i) nom-y-dir	l) num1_art

- 7- Supongamos que la versión de C puede reconocer sólo los ocho primeros caracteres del nombre de un identificador, aunque los nombres de identificadores pueden ser de una longitud arbitraria. ¿Cuáles de los siguientes pares de nombres de identificadores se consideran como idénticos y cuáles se distinguirán?

a) nombre, nombres	c) identificador_1, identificador_2	e) respuesta, RESPUESTA
b) direccion, direcciones	d) lista1, lista2	f) num1, num_1

- 8- Determinar cuales de los sgtes. valores numéricos son constantes válidas. Si es una constante válida especificar si es entera o real.

a) 0.5	c) 9.3e12	e) 12345678
b) 27,822	d) 9.3e-12	f) 127.09

- 9- Determinar cuales de las sgtes. son constantes de carácter válidas.

a) 'a'	c) '\n'	e) '\\'	g) 'T'	i) 'xyz'
b) 'S'	d) '/n'	f) '\a'	h) '\0'	j) "a"

- 10- Determinar cuales de las sgtes. son constantes de cadena de caracteres válidas.

- '8:15 P.M.'
- "rojo, blanco y negro"
- "Nombre:"
- "Capitulo 3 (cont\'d)""1.3e-12"
- "2000 ROSARIO"
- "El profesor dijo: "Por favor, presten atención""

- 11- Escribir las declaraciones apropiadas para cada grupo de variables:

Variables enteras: p, q
 Variables de coma flotante: x, y, z
 Variables de carácter: a, b, c
 Variable entera larga: cont
 Variable entera corta: indicador
 Variable entera sin signo: cliente_num
 Variables de doble precisión: bruto, impuesto, neto

- 12- Escribir las declaraciones apropiadas y asignar los valores iniciales dados para cada grupo de variables

Variables de coma flotante: a = -8.2, b = 0.005
 Variables enteras: x = 129, y = 87, z = -22
 Variables de carácter: c1 = 'w', c2 = '*'
 Variable de carácter: eol = carácter de nueva línea

- 13- Dentro de una función, si 2 variables, una local y una global, tienen el mismo nombre:

- La global prevalece sobre la local
- La local prevalece sobre la global
- Se producirá un error

- 14- Detectar y justificar los errores en cada uno de los siguientes programas:

a.

```
main()
{
/* Aquí va el cuerpo del programa */
Printf( "Hola mundo\n" );
}
```

b.

```
#include <stdio.h>
```

```
main()
{
int número;

número = 2

printf( "El valor es %i" Numero );
}
```

c.

```
#include <stdio.h>
```

```
main()
{
int a, c;

a = 5;
c += a +5;
}
```

d.

```
#include <stdio.h>
```

```
main()
{
ClrScr();
gotoxy( 10, 10 )
printf( Estoy en la fila 10 columna 10 );
return 0;
}
```

- 15-Cuál es el resultado de cada una de las siguientes expresiones

a) $!(3 > 3 \ \&\& \ 5 != 4 \ || \ 9 == 9)$ _____

b) $!(7 > 8) \ \&\& \ 7 > 8 \ || \ 7 > 8$ _____

c) $13/13*2*5+5\%2$ _____

d) $x = (y = 3, y + 1)$ _____

- 16- Construir una expresión para cada una de las siguientes condiciones:

a) El valor de la variable num no está entre 1 y 5

.....

b) El valor de la variable car no es el carácter 'A' ni el carácter 'Z'.

.....

c) La variable num está entre 0 y 9 pero no es 3.

.....

17- Dadas las sgtes declaraciones y asignaciones: **int a = 2, b =3, c = 4**; evaluar las sgtes. expresiones y determinar que quedará almacenado en **result** en cada caso.

1. result = a * b % c;
2. result = 3 * (a -= 1) + 1 % a;
3. result = (a > b) || (b = c);

18- Que resultado se obtiene de evaluar los siguientes segmentos de programa:

a. main()

```
{
    int a = 3, b = 5, c = 4;
    if (a < b > c) printf ("Correcto");
    else printf("Incorrecto");
}
```

b. main()

```
{ int a = 10, b= 3;
(a % b == 0 )? printf(" Azul\n"): printf(" Rojo\n");
}
```

c. main()

```
{
    int a = 10, b= 5;
    char c;
    c = (a % b == 0 )? 'A' : 'B';
    printf(" %c ", c);
}
```

d. main()

```
{
    int a = 10;
    float c, d, b = 4;
    c = b / a;
    d = (int)b / a;
    printf( " %f \n ", c);
    printf( " %f \n ", d);
    printf( " %g \n ", d);
    printf( " %g \n ", c);
}
```

```
e. main()
{
    char c ; int i ; float f ;
    c = i = f = 60 + 5.8;
    printf ( " %c %d %f\n", c, i ,f) ;
}
```

```
f. main()
{
    char c ; int i ; float f ;
    c = i = f = 'A' + 5 ;
    printf( " %c %d %f", c, i, f);
}
```

```
g. main()
{
    char c ; int i ; float f ;
    c = i = f = 'A' + 5.9 ;
    printf ( " %c %d %f\n", c, i, f) ;
}
```

```
h. main()
{
    char c ; int i ; float f ;
    c = i = f = 'A' + 3.3 + 5;
    printf( " %c %d %f", c, i, f) ;
}
```

```
i. main( )
{
    int a =1, b = 5;
    printf( "%d\t", a = 2);
    printf( "%d\n", 5 / 2 * 3 + 8 % 5 + b);
}
```

```
j. main( )
{
    int a =1, b = 5;
    printf( "%d\t", a = 1);
    printf( "%d\n", b + 7 / 3 * 3 + 7 % 5 );
}
```

```
k. #include <conio.h>
#include <stdio.h>
main()
{
    int a, b, c;
    a = 5;
    b = ++a;
    c = ( a + 5 * 2 ) * ( b + 6 / 2 ) + ( a * 2 );
    printf( "%i, %i, %i", a, b, c );
}
```

19- ¿Qué valor contendrán las variables **s** y **c** luego de ejecutarse las secuencia de asignaciones que se mencionan en cada caso?

a)	$c = 0;$	b)	$c = 0;$	c)	$c = 4;$
	$s = (++c) + 1 ;$		$s = c ++ + 1;$		$c * = 2 ;$
	$s = c + 1;$				

20- ¿Qué valor queda en cada variable involucrada, después de ejecutar el enunciado que se indica, suponiendo que antes de ello todas las variables contienen el valor 5?

a) $product * = x ++ ;$ b) $result = ++x + x;$

- 21- Escribir al menos de tres formas distintas la operación incrementar en **1** la variable **cont**.
- 22- Completar las líneas de punto
- a) El operador..... permite realizar una operación si una condición es verdadera y otra si es falsa .
 - b) El operador..... devuelve el resto de la división entera.
 - c) Las variables definidas fuera de toda función se denominan.....
 - d) Una expresión lógica es verdadera si.....
 - e) Toda sentencia termina en.....
 - f) Un programa escrito en lenguaje C esta íntegramente formado por
 - g) Todo programa en C comienza su ejecución en la función.....
- 23- Responder a cada una de las siguientes afirmaciones con V (verdadero) o F (falso) según corresponda. Justificar.
- a. El operador % puede ser utilizado con cualquier tipo de operandos.
.....
 - b. En la expresión `a= ('4' > 4)` la variable a asume el valor 1.
.....
 - c. La expresión `(x > y || a < b)` es verdadera ya sea que `(x > y)` es verdadero o `(a < b)` es verdadero
.....
 - d. Una variable local puede ser definida en cualquier lugar dentro de una función.
.....
- 24- Escribir 2 sentencias diferentes para realizar:
Dado un numero imprimir “n” si el número es < 0 y “Z” si el número es > 0 .
- 25- Asignarle a la variable **var** el contenido de la variable **res** y luego disminuir **res** en **1**(uno).

- 26- Asignarle a la variable **p** un **1 si p es impar o un 0 si p es par**, esto en el caso de que **p sea mayor que 20**, sino asignarle un **6**.

FUNCIONES DE ENTRADA - SALIDA

- 27- Escribir los enunciados necesarios para ingresar por teclado las variables:
- x de tipo entera e y de tipo flotante, separadas por una ,
 - x de tipo caracter e y de tipo entera, separadas por una /
 - x de tipo entero largo e y de tipo caracter, separadas por un blanco
- 28- Escribir los enunciados necesarios para mostrar:
- La mesa numero num esta servida
 - Hoy es el dia nro num de la semana
 - En la columna nro num hay un error
 - Los numeros son: n1, n2, n3
 - El importe de la factura es num
- 29- Escribir un programa que ingrese un carácter y muestre en pantalla el número decimal que le corresponde en la tabla ASCII
- 30- **Decir si scanf(), printf(), getchar(), clrscr() son:**
- | | | |
|--|-----------|-----------|
| a) palabras reservadas del lenguaje C | SI | NO |
| b) funciones cuyos prototipos se encuentran en algún archivo de cabecera cuya extensión es .h | SI | NO |
| c) funciones creadas por el programador | SI | NO |
- 31- Leer una cadena de caracteres y un valor entero ambos separados por un * almacenándolos en las variables **cad** y **num** respectivamente.
- 32- Ingresar dos números enteros y luego presentar el siguiente menú de opciones:
- | | |
|----|-------------|
| 1- | SUMAR |
| 2- | RESTAR |
| 3- | MULTIPLICAR |
| 4- | DIVIDIR |
- A esto el usuario debe responder con la opción correspondiente a la operación que desee hacer entre los números y el programa debe devolver el resultado.
- 33- Escribir un programa que ingrese un entero y muestre en forma tabulada el número, su cuadrado y su cubo.
- 34- Escribir un programa que lea el radio de un círculo y que imprima el diámetro del mismo, su circunferencia y su área. Utilice el valor constante 3.14159 para "PI" y defina al mismo como constante simbólica.

- 35- Escribe un programa que borre la pantalla y escriba en la primera línea su nombre y en la segunda su apellido:

- 36- Escriba un programa que borre la pantalla y muestre el texto "estoy aquí" en la fila 10, columna 20 de la pantalla.

- 37- Cuál será a salida del siguiente trozo de programa?

```
#include <stdio.h>

int main()
{
    char nombre[8] = "Timoteo";
    int hermanos = 2, sobrinos = 4;

    printf( "%s tiene %d hermanos y %d sobrinos.",
            nombre, hermanos, sobrinos );

    return 0;
}
```

- 41- Escribir un programa que contenga la siguiente instrucción

```
printf( "diez\tveinte\ttreinta" );
```

y determinar que se obtiene por pantalla

- 43- Hacer el algoritmo que nos permita introducir un número por teclado y nos informe si es positivo o negativo

- 44- Hacer el algoritmo que nos permita introducir un número por teclado y nos informe si es par o impar

- 45- Escribir el algoritmo que me permita leer un número decimal que representa una cantidad de grados Celsius y convierta dicho valor a la cantidad equivalente en grados Fahrenheit. La salida del programa puede ser de la siguiente forma: 100 grados Celsius son 212 grados Fahrenheit

- 46- Diseñar el algoritmo necesario para que habiéndose leído el valor de 2 variables NUM1 y NUM2 se intercambien los valores de las variables, es decir que el valor que tenía NUM1 ahora lo contenga NUM2 y viceversa

- 47- La fecha de Pascua corresponde al primer domingo después de la primera luna llena que sigue al equinoccio de primavera, y se calcula con las siguientes expresiones:

```
A = año mod 19
B = año mod 4
C = año mod 7
D = (19*A + 24) mod 30
E = (2*B + 4*C + 6*D + 5) mod 7
```


$$N = (22 + D + E)$$

En el que N indica el número del día de marzo (o Abril si $N > 31$) correspondiente al domingo de Pascua. Realizar un programa que calcule esa fecha para los años 2008 y 2009.

48- Dada la siguiente llamada a la función scanf()

```
scanf ("%3d %3d %3d",&a,&b,&c);
```

- Si los datos se introducen: 1 2 3 las asignaciones que se llevan a cabo son:
- Si se introducen 123 456 789 las asignaciones son:
- Si se introduce 123456789 las asignaciones son:
- Si se introduce 1234 5678 9 las asignaciones son:

49- Escribir un enunciado C para conseguir lo siguiente:

- Ingresar un texto carácter a carácter hasta que se ingrese '^Z', sin almacenar dicho texto.
- Ingresar un texto carácter a carácter hasta que se ingrese '#', sin almacenar dicho texto.

SENTENCIAS DE CONTROL DE FLUJO

50- Decir que muestra cada una de las instrucciones printf.

```
#include <stdio.h>
#include <conio.h>
main( )
{
    int  expresión, x = 1;
    expresion = ! ( (1 == x) && 4)
    if (expresión )
        printf("El resultado de la expresión es verdadero");
    else
        printf("El resultado de la expresion es falso");
    x = 4;
    for( ; x < 10; x += 3);
    printf("%d", x / 2);
    x = 10;
    do
    {
        printf("%d", x--);
    } while ( x >5);
}
```

- 51- Mostrar una tabla que contenga los valores de peso y costo, variando el peso desde 2 hasta 20 con un incremento de 6, si el costo mínimo es de \$50 y se incrementa en \$15 cada 6 kg. Utilizar un bucle **for** y constantes simbólicas donde se considere útil
- 52- Incrementar una variable entera **j** desde 0 hasta **n** y luego mostrar el resultado. Hacerlo utilizando 3 bucles diferentes.
- 53- En la expresión **for(i=1; i==10; i+=2);** cuántas veces se evalúa la condición
- 54- Ingresar n números y calcular su media ($\text{media} = (\text{nromayor} + \text{nromenor}) / 2$)
- 55- Escribir el algoritmo necesario para calcular y mostrar el cuadrado de un número ingresado por teclado. El número debe ser mayor que cero, en caso de error que aparezca el mensaje "ERROR, el número debe ser mayor que cero". El programa termina su ejecución cuando se ingresa un 0
- 56- De 10 números ingresados indicar cuantos son mayores que cero y cuantos son menores que cero.
- 57- Leer sucesivamente de teclado, hasta que aparezca un número comprendido entre 1 y 5. Desarrollar el algoritmo usando la función :

a) *getchar()*b) *scanf()*
- 58- Codificar en C un programa que lea 20 caracteres indique cuantas "a" se ingresaron, cuantas "e, i, o, u"
- 59- Realizar un algoritmo que permita ingresar un número correspondiente a los días de una semana y muestre el nombre del día. Que se permita trabajar hasta que el usuario indique lo contrario.
- 60- Hacer el algoritmo que imprima los números pares entre el 1 y el 100 a razón de 6 por línea separados por 3 blancos
- 61- Hacer el algoritmo que imprima los números del 100 al 0 en orden decreciente con formato: 00100 00099 00098.....00001 00000 5 por línea
- 62- Diseñar un algoritmo que genere e imprima los múltiplos de 3 desde 3 a 99 y luego muestre su sumatoria.
- 63- Codificar en C un programa que muestre los múltiplos de 2 y de 3 y de ambos comprendidos entre 1 y 100
- 64- Hacer un algoritmo que imprima todos los números pares que hay desde la unidad hasta un número que introducimos por teclado. El programa debe poder ejecutarse mientras el usuario lo requiera.

- 65- Codificar en C un programa que muestre los números primos comprendidos entre 0 y 100
- 66- Codificar en C un algoritmo que permita ingresar 10 números, ninguno de ellos igual a cero. Se pide sumar los positivos, obtener el producto de los negativos y luego mostrar ambos resultados.
- 67- Calcular la media de los números no negativos de una lista de **n** números ingresados por teclado (**n** se ingresa por teclado)
- 68- Dados 3 números donde el primero y el último son límites de un intervalo, indicar si el tercero pertenece a dicho intervalo
- 69- Realizar la tabla de multiplicar de un número entre 0 y 10 de forma que se visualice de la siguiente forma:
 4x1= 4
 4x2= 8

- 70- Hacer el algoritmo que luego de ingresar 2 números por teclado, imprima los números pares que hay entre ambos empezando por el más pequeño, contar cuántos números hay y mostrarlo.
- 71- Hacer un algoritmo que imprima el mayor y el menor de una serie de 5 números que vamos introduciendo por teclado.
- 72- Escribir un programa que determine si un año es bisiesto. Un año es bisiesto si es múltiplo de 4. Los años múltiplos de 100 no son bisiestos, salvo si ellos también son múltiplos de 400 (2000 es bisiesto pero 1900 no)
- 73- Escribir un programa que solicite el ingreso de un número entero y a partir de él, genere una cuenta regresiva hasta cero, imprimiendo en el mismo lugar los sucesivos valores hasta el último luego de lo cual emitirá un pitido.
- 74- Escribir un programa que genere y muestre en pantalla la tabla ASCII.
- 75- Ingresar un texto de caracteres utilizando la función `getchar()`. Indicar la cantidad de caracteres, palabras, y líneas que lo forman. Mostrar lo pedido con carteles aclaratorios. Considerar como separadores de palabras válidos: espacio, tabulador y enter y tener en cuenta que contar palabras no es contar cantidad de separadores.
- 76- Escribir un programa para revisar los errores de sintaxis rudimentarios de un programa en C.(llaves, paréntesis, corchetes, comilla y apóstrofes)
- 77- Desarrollar un algoritmo que permita el ingreso de un número y determine si el mismo es primo, dicho algoritmo debe poder repetirse tanta veces como se considere necesario.

ARREGLOS Y CADENAS - FUNCIONES

- 78- Dado el siguiente programa indicar que resultado se obtiene.

LIC M. Jäger – LIC S. Martín

```
#include<stdio.h>
main()
{ int vec[]={ 1,2,-2,1,3,-1,5,10}, i, var;
  var=1;
  for( i = 0; i <= 7; ++i )
    if(vec[i]>0) var = var * vec[i];
  printf ( “ %d”, var);}
```

79- Dada la siguiente declaración e inicialización del arreglo A

```
{ int A[5] = { 1,2,3,4,5}
```

Responder:

Valor de A[0] =

Valor de A[4] =

80- Dado el siguiente programa indicar que resultado se obtiene.

```
include<stdio.h>
main()
{ int vec[]={ 1,2,-2,1,3,-1,5,10}, i, var;
  var= i = 0;
  do{ var = var + vec[i];
    ++i;
  }while (vec[i] >0);
  printf ( “ %d”, var);}
```

81- Dado el siguiente programa indicar que resultado se obtiene.

```
#include<stdio.h>
main()
{ int vec[]={ 1,2,-2,1,3,-1,5,10,-5,2,3}, i, var;
  var= i = 0;
  while( vec[i] < 10)
  { var = var + vec[i];
    ++i;
  }
  printf ( “ %d”, var);}
}
```

82- Escribir un programa que permita el ingreso de 1000 números, los almacene en un arreglo, luego ingrese otro número y determine entre los anteriores cuales son mayores o iguales al último número ingresado. Con los mayores calcular su sumatoria y mostrar:

Nº Ingresado

LA SUMATORIA DE LOS NÚMEROS MAYORES QUE número ES sumatoria

Resultado

- 83- Escribir un programa que ingrese y almacene 1000 números y que los muestre ordenados según un indicador de criterio de ordenamiento que también debe ingresarse por teclado.

- 84- Escribir un programa que cargue un arreglo bidimensional numérico e indique la fila y la columna en la que se encuentra almacenado el número de mayor valor absoluto.

- 85- Escribir un programa que ingrese 10 números enteros, los cargue en un arreglo unidimensional, busque el mayor de ellos, calcule el promedio y muestre:

LOS NÚMEROS INGRESADOS SON :

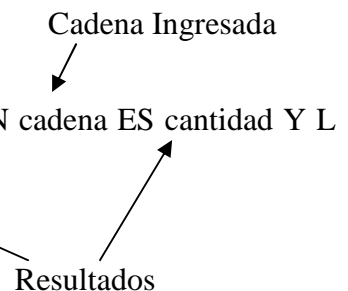
EL MAYOR DE ELLOS ES:

SU PROMEDIO ES :

- 86- Escribir un programa que cree un arreglo de 26 elementos y que además genere y guarde en ellas 26 las letras minúsculas del alfabeto.

- 87- Escribir un programa que ingrese una cadena, cuente cuantos de sus elementos no son dígitos, cuantos de ellos son blancos y muestre:

LA CANTIDAD DE NO DÍGITOS QUE APARECEN EN cadena ES cantidad Y L
CANTIDAD DE BLANCOS ES cant_blanco



```

graph TD
    CI[Cadena Ingresada] --> R[Resultados]
    R --> O[LA CANTIDAD DE NO DÍGITOS QUE APARECEN EN cadena ES cantidad Y L  
CANTIDAD DE BLANCOS ES cant_blanco]
    
```

- 88- Sumar números en un vector de 10 elementos enteros hasta que el usuario lo desee. Cuando finalicen las sumas visualizar por pantalla lo siguiente:

Elemento 0 cantidad total

.....

Elemento 9 cantidad total

- 89- Leer dos vectores A y B de dimensión 10. Generar un tercer vector C de 100 elementos donde la componente $C[i]$ sea igual al mínimo valor de $A[i]$ y $B[j]$.
Mostrar los tres vectores

- 90- Leer 20 números enteros positivos. Mostrar el vector tal como fue ingresado y luego mostrar el vector ordenado en forma decreciente.

91- Leer 20 números enteros (positivos y negativos) distintos de cero. Mostrar el vector tal como fue ingresado y luego mostrar los positivos ordenados en forma decreciente y por último mostrar los negativos ordenados en forma creciente.

92- Se dispone de dos cadenas de caracteres **cad1** y **cad2** y se ejecutan las sgtes. funciones:

- **strcpy(cad1, "ANTONIO");**
- **strcpy(cad2, "ANTENA");**

Para la expresión **strcmp(cad1, cad2)**; ¿Cuál de las sgtes. afirmaciones es cierta?

a) Lo que devuelve la expresión dependerá de la dimensión de cad1 y cad2.

b) La expresión devuelve un n° menor que cero.

c) La expresión devuelve un n° mayor que cero.

d) Lo que devuelve la expresión dependerá de la dimensión de cad1.

e) Lo que devuelve la expresión dependerá de la dimensión de cad2.

93- Decir cual será la salida de los siguientes programas.

a)

```
main()
{
    char cad1[ ] = "Examen";
    char cad2[ ] = "Parcial";
    printf("%s\n", strrev(cad2));
    printf("%d", strlen(cad1));
}
```

b)

```
main()
{
    char *cadena = "líneas";
    strcpy(cadena, "5, 89, 123,2");
    strrev(cadena);
    printf ( " %s \n", cadena);
    printf ( " %c \n", *cadena);
}
```

94- Hacer un programa que ingrese dos cadenas y:
 . Determine cuál de ellas es mayor en orden alfabético.
 . Concatene ambas cadenas.
 . Determine cuál es mas larga.
 . Invierta la primera ingresada.

El programa deberá mostrar todas las salidas con carteles aclaratorios

LA MAYOR ES

LAS CADENAS CONCATENADAS QUEDAN.....

LA CADENA MÁS LARGA ES.....

LA CADENA cadena INVERTIDA SE LEE

- 95- Programar una función que, dada una cadena y un carácter, retorne la subcadena que comienza con dicho carácter o NULL en caso de que el caracter no exista en la cadena
- 96- Ingresar una cadena de caracteres e indicar si la misma es un palíndromo. Realizar esta operación cuantas veces se desee.
- 97- Dado un texto presentar todas las palabras que lo componen ordenadas alfabéticamente. Optimizar la memoria requerida.
- 98- Escribir un programa que encuentre el número de veces que una palabra dada ocurre en un texto
Una salida podría ser:
La palabra es "el"
El texto es "el perro, el gato y el canario"
La palabra ocurrió 3 veces.
- 99- Escribir un programa en **C** que solicite el ingreso de un texto, luego ingrese una letra y muestre, ordenadas alfabéticamente, todas las palabras del texto que comienzan con dicha letra. El ingreso de la letra se podrá repetir tantas veces como se desee. No utilizar variables globales.
- (Cant. Max. Caracteres: 1500 – Cant. Max. Palabras: 100 – Long.Max. Palabra: 15 car.)
- 100- Escriba un programa que inserte un carácter en una determinada posición de una cadena.
- 101- Escribir el correspondiente programa en código **C**:
Ingresar por teclado un texto, el final del mismo se dará con **EOF**.
Presentar en pantalla el siguiente menú de opciones:

- 1- CHARACTER**
- 2- COMA**
- 3- LETRA**
- 4- FIN**

Y para la opción:

- 1-CARÁCTER**, ingrese un carácter, reemplace los espacios en blanco por el mismo y muestre el nuevo texto.
- 2-COMA**, elimine todas las comas y muestre el nuevo texto.
- 3-LETRA**, ingrese una letra, cuente todas las palabras del texto que empiezan con la misma y muestre apropiadamente el resultado obtenido

es	3 veces
sintaxis	2 veces
y	2 veces

OTROS TIPOS DE DATOS - ESTRUCTURAS

104- Dado el siguiente segmento de programa:

```
struct dato {int codigo;
             char descripcion[20];
             float precio;
             }a;

main()
{ struct dato b;
  .....
  .....
}
```

- a) ¿Son **a** y **b** dos variables que tienen el mismo tipo de dato? **SI - NO**
 b) ¿La variable **a** es de ámbito local? **SI - NO**

105- Declarar una estructura de nombre **alumnos** para almacenar la siguiente información:

- nombre del alumno
 - n° de legajo
 - domicilio
 - turno,
- de cada uno de los 300 alumnos de un establecimiento.
 Luego de declararla asignar los sgtes. datos en la 1ª estructura: Juan Gomez , 123245, San Juan 1056, M

106- Dada la siguiente declaración:

```
struct asistencia {
    char turno;
    int dia [6];};
```

- a) Declarar una variable alumnos de tipo estructura **asistencia**.
 b) Asignar al campo turno de la variable alumnos una **M** y un **0** a la primera componente del campo dia.

107- De los alumnos de una escuela se registra su apellido y nombre y su altura. Diseñar un algoritmo que indique el nombre del alumno más alto y su altura (sólo uno es el más alto).

108- Diseñar un algoritmo que permita calcular el promedio de un alumno sabiendo que se ingresa por alumno la nota de 6 materias y su nombre. No se sabe la cantidad de alumnos. Se pide mostrar:

NOMBRE DEL ALUMNO: PROMEDIO

- 109- Realizar un programa que permita obtener las raíces reales de la ecuación de segundo grado x^2+bx+c .
- 110- Se ingresa nombre, sexo y edad de 100 personas. Se pide averiguar la cantidad de mujeres que tienen entre 20 y 30 años y cuantos hombres son menores a 37 años.
- 111- Ingresar los datos de alumnos (46 como máximo): legajo, nombre, fecha de nacimiento. Según se elija, informar los datos de los alumnos ordenados por Legajo o Fecha de Nacimiento
- 112- Se dispone de registros con los sgtes. datos: código (5 caracteres), nombre (20), domicilio (20), saldo.
Se solicita cargar dichos datos (500 registros como máximo) y luego satisfacer un esquema de consulta interactiva donde al ingresar un código se informan los datos restantes correspondientes.
- 113- Se ingresa el apellido, la nota y el legajo de los 30 alumnos de un curso. Realizar un programa que permita modificar los datos ingresados hasta que el usuario lo determine. Para ello se ingresa el legajo y la nota a modificar. Si no se encuentra el legajo se deberá mostrar un mensaje de error. Finalmente mostrar los datos ordenados por legajo
- 114- En un comercio se carga el nombre, precio y cantidad de 50 artículos.
Hacer un programa que me permita ingresar los datos y luego mostrar ordenado por nombre los artículos que tienen menos de 10 unidades en stock.
- 115- Realizar un programa que por medio de un menú de opciones permita hacer ABM de los datos de un máximo de 20 personas. Los datos a ingresar son Nombre y fecha de nacimiento.
- 116- Un profesor de matemática de un establecimiento educativo registra de cada alumno N° de legajo, nombre y promedio. Según el promedio desea saber cuantos alumnos aprobaron (promedio mayor o igual a 7), cuantos rinden en diciembre (promedio menor a 7 y mayor o igual a 4) y cuantos rinden examen en marzo (promedio menor a 4). Además desea conocer el N° de legajo y nombre del alumno con mejor promedio.
- 117- De los 100 empleados de una fábrica se registra: número de legajo (coincide con el número de índice), edad y salario. Se pide:
- Ingresar los datos correlativamente
 - Calcular la edad promedio informando número de legajo de los empleados cuya edad supere el promedio.

118- De los 100 alumnos de una escuela se ingresa:

nota del 1° trimestre, nota del 2° trimestre, nota del 3° trimestre

El orden de ingreso coincide con su número de legajo que es de 1 a 100

Luego del registro el programa debe preguntar al usuario si desea corregir alguna nota, para lo cual se debe ingresar:

N° de legajo

Trimestre a corregir (1,2 o 3)

Nueva nota

El programa debe mostrar la nota anterior correspondiente al trimestre indicado y preguntar si confirma la modificación. Si contesta “SI” se registra la nueva nota, caso contrario pregunta si desea continuar. Cuando no se registren más modificaciones se deberá listar:

N° DE LEGAJO	1° TRIMESTRE	2° TRIMESTRE	3° TRIMESTRE	PROMEDIO
XXXX	XX	XX	XX	XX.XX

119- Una compañía de aviación tiene 4 destinos (numerados de 1 a 4) con 3 vuelos cada uno (numerados de 1 a 3). Se ingresa la información de las plazas disponibles en cada uno de los 12 vuelos. Posteriormente llegan pedidos de pasajes de los que se ingresa:

N° de pedido

N° de destino

N° de vuelo

Cantidad de pasajes requeridos

Si la cantidad de pasajes pedidos es superior a la disponibilidad existente en ese vuelo y para ese destino, se rechaza el pedido indicando la leyenda: NO HAY PLAZAS DISPONIBLES.

Se pide informar :

- la cantidad de pasajes sobrantes en cada vuelo
- los N° de pedidos rechazados con la cantidad de pasajes solicitados

120- Una fábrica tiene 30 obreros con legajos de 1 a 30. Por cada obrero se registran los movimientos que intervienen en la liquidación informando:

N° de legajo

Sueldo básico

Bonificación por antigüedad

Un descuento por jubilación que representa el 16 % del sueldo básico

Un descuento por obra social que representa el 4,5 % del sueldo básico

Se pide:

- Listado como el siguiente:

N° de legajo	Sueldo básico	Jubilación	Obra social	Antigüedad	Sueldo neto

- Indicar cual fue el legajo del o los obreros con mayor bonificación de antigüedad y cual fue esa cantidad
- Listado de todos los obreros ordenado por sueldo neto

121- Leer 10 estructuras del arreglo "empleado" que contiene la siguiente información:

nombre

dirección (calle, número, localidad)

edad

sexo (M = masculino, F = femenino)

a) listar los empleados menores de 25 años

b) listar los empleados que viven en Avellaneda

c) listar los empleados que viven en Avellaneda menores de 25 años

d) ordenar el vector por nombre y listarlo

122- Armar 2 estructuras

producto:

código

precio unitario

descripción

stock

cliente:

código

datos (nombre y dirección)

monto acumulado de compras

a) Cargar los productos (100) y los clientes (45). Ambas cargas deben realizarse una sola vez.

b) Ingresando código del cliente, código del producto y cantidad vendida, confeccionar la factura correspondiente realizando los siguientes controles:

- verificar que cantidad vendida no supere existencia en stock

- debitar stock en cada venta

- actualizar monto acumulado de compras de cada cliente

c) Mostrar por pantalla la factura

d) listar los clientes cuyo monto acumulado de compras supere \$1000

123- En una biblioteca se registran los libros existentes, informándose:

-CODIGO DE LIBRO (De 1 a 100)

-CANTIDAD DE EJEMPLARES DISPONIBLES

-AUTOR

-TITULO

Luego de registradas las existencias se visualiza un menú que permite al bibliotecario elegir alguna de las siguientes opciones:

OPCION 1: PRESTAMOS DE LIBROS

El bibliotecario debe ingresar :

CODIGO DE LIBRO (De 1 a 100, caso contrario informar error y volver a ingresar)

El programa debe mostrar el autor y el título y sólo se presta el libro si la cantidad de ejemplares es mayor a 1.

Si el usuario confirma el préstamo restar 1 a la cantidad de ejemplares disponibles.

Si la cantidad de ejemplares es 1, se mostrar la leyenda "Solo queda el ejemplar de lectura en sala" y no se registrar el préstamo.

OPCION 2: DEVOLUCION DE LIBROS

El bibliotecario debe ingresar :

CODIGO DE LIBRO (de 1 a 100, caso contrario informar error y volver a ingresar)

El programa debe mostrar el autor y el título y si el usuario confirma, registrar la devolución sumando 1 a la cantidad de ejemplares disponibles.

OPCION 3: FIN

Significa que no se sigue trabajando con el menú de opciones y que antes de finalizar debe informar lo siguiente:

Listado ordenado en forma decreciente por cantidad de veces que se prestó un libro:

TITULO	AUTOR	VECES PRESTADO
XXXXXXXXXXXXXX	XXXXXXXXXXXXXX	XXX

124- Una empresa registra por cada artículo en existencia:

CODIGO DE ARTICULO (de 1 a 30)

DESCRIPCION DEL ARTICULO

CANTIDAD EN STOCK

PRECIO UNITARIO

Luego del registro de stock se visualizar un menú de opciones que permita al usuario elegir alguna de las siguientes:

OPCION A: registrar compras y ventas, para lo cual debe informar :

CODIGO DE ARTICULO (de 1 a 30 , caso contrario informar error y volver a ingresar)

CODIGO DE MOVIMIENTO (1 o 2 caso contrario se informa error vuelve a ingresar). El código 1 indica una venta y 2 una compra.

CANTIDAD (comprada o vendida de acuerdo al código del movimiento)

Preparar esta opción para que se continúe trabajando con ella hasta que el usuario decida terminar y pasar al menú principal.

OPCION B: consultar el precio unitario de un artículo, de modo tal que al ingresar un código de artículo(de 1 a 30 , caso contrario informar error y volver a ingresar), el sistema debe mostrar la descripción del artículo y el precio unitario.

OPCION C: consultar existencia en stock de un artículo de modo tal que al ingresar el código de artículo(de 1 a 30 , caso contrario informar error y volver a ingresar), el sistema debe mostrar la descripción del artículo y su existencia.

OPCION D: finalizar la ejecución del programa.

PUNTEROS

125- Decir cual será la salida:

```
a  #include<stdio.h>
    main()
    {
        int arr[10] = {23, 5, 98, 65, 3, 55, 73, 9, 21, 85}, *p;
        p = arr;
        printf ( " %d\n", arr[(p + 7)]);
        printf ( " %d\n", *arr + 3);
        printf ( " %d\n", *p++);
        printf(" %d\n", *(arr + 1));
```

```
printf(" %d\n", (*p)++);
}
```

```
b #include<stdio.h>
main()
{
    char *p = " esto es una prueba";
    printf ( p );
}
```

- 126- Suponiendo que **p** es un puntero a float que actualmente apunta a la posición 100 y que los floats tienen 4 bytes de longitud ¿cuál es el valor de P después de que se haya ejecutado la siguiente sentencia?
- ```
p = p + 2;
```
- 127- Escribir un programa en C que con un bucle for cuente de 0 a 10, mostrando los números en la pantalla utilizando un puntero.
- 128- Dado el sgte. Trozo de programa:

```
char u, v = 'A';
char *pu, *pv = v;
.....
*pv = v + 1;
u = *pv + 1;
pu = &u;
```

y suponiendo que el valor asignado a **u** se almacena en la dirección F8C y el valor asignado a **v** se almacena en la dirección F8D, decir:

- a ¿Qué valor es representado por &v?
  - b ¿Qué valor es asignado a pv?
  - c ¿Qué valor es representado por \*pv?
  - d ¿Qué valor es asignado a u?
  - e ¿Qué valor es representado por &u?
  - f ¿Qué valor es asignado a pu?
  - g ¿Qué valor es representado por \*pu?
- 129- Escribir un programa en C que declare un puntero a char y lo inicialice con su nombre y luego lo muestre, utilizando punteros, carácter a carácter hasta que detecte el carácter nulo de final de cadena ('\0')
- 130- Escribir un programa en C que genere 100 números, los almacene en un arreglo y luego muestre por pantalla el número que se encuentre en la posición **n** (siendo **n** un número ingresado por teclado) utilizando un puntero.
- 131- Escribir un programa en C que inicialice un arreglo bidimensional con los números del 1 al 6 y luego los muestre utilizando punteros.
- 132- Escribir un programa que no pare de leer cadenas hasta que se introduzca “**fin**”.
- 133- Crear una función de nombre **micopy** que me permita copiar una cadena en otra utilizando punteros.

- 134- Crear una función de nombre **compara** que me permita comparar alfabéticamente dos cadenas y me retorne: “SON IGUALES”, “LA PRIMERA ES MAYOR” O “LA PRIMERA ES MENOR” según corresponda. Utilizar punteros.
- 135- Escribir un programa en C que me permita ordenar según un indicador de criterio de ordenamiento que debe ingresarse por teclado un arreglo de **n** números generados al azar y que los muestre. El programa deberá poder ejecutarse tantas veces como el operador lo desee. La cantidad de números a generar variará de una ejecución a otra y deberá reservarse solamente la memoria necesaria para cada ejecución.
- 136- Programar una función que, dado un arreglo de enteros con 0 (cero) como marca de fin, devuelva un puntero al primer valor que no esté ordenado en forma creciente. Si el arreglo está ordenado retornará NULL Ej: Si el arreglo es: 44, 58, 65, 13, 33, 0 la función deberá retornar un puntero al cuarto elemento.