

Trabajo Práctico 4

Temas:



- ❖ Cadenas de caracteres.
- ❖ Arreglos de cadenas de caracteres.

ES OBLIGATORIO UTILIZAR NOMBRES DE VARIABLES Y DE FUNCIONES SIGNIFICATIVOS

Problemas propuestos

Cadenas de Caracteres

1. Solicitar el nombre de una persona y almacenarla en una **cadena de caracteres** al usuario y luego mostrarla por pantalla. Emplear las diferentes funciones para lectura y presentación por pantalla y las opciones de juegos de inspección para `scanf()`, formato de salida para `printf()`.
2. Solicitar al usuario el ingreso de una cadena de caracteres que puede contener solamente los dígitos 0-9, luego presentarla por pantalla. Realizar el ingreso usando `scanf()` con las opciones de juego de inspección.
3. Solicitar al usuario el ingreso de una cadena de caracteres que puede contener solamente caracteres alfabéticos, luego presentarla por pantalla. Realizar el ingreso usando `scanf()` con las opciones de juego de inspección.
4. Solicitar al usuario el ingreso del apellido y nombre de una persona y almacenarlos en una **cadena de caracteres**, luego mostrarla por pantalla. Realizar la lectura y presentación por pantalla empleando todas las funciones vistas: `scanf()`, `fscanf()`, `fgets()`, `printf()`, `fprintf()` y `fputs()`. Analizar las diferencias y qué sucede con el salto de línea (enter) y los espacios.
5. Ingresar una cadena de caracteres, contabilizar cuántos caracteres alfabéticos, numéricos y de otro tipo hay en la misma. Mostrar la cadena leída y las cantidades encontradas.
6. Ingresar una cadena de caracteres que contenga dos o más palabras separadas por un espacio en blanco. Luego presentar por pantalla la cadena ingresada y la cantidad de palabras que posea. Ejemplo: Se ingresa -> Tercer evaluativo de Programación I -> Muestra: 5 palabras.
7. Ingresar dos cadenas de caracteres y presentarlas por pantalla en el orden en que se encuentran en el diccionario. Usar **`strcmp()`**. Ejemplo se ingresan "Sanchez Mateo" y "Sanchez Mariana" y se debe mostrar "Sanchez Mariana", "Sanchez Mateo"
8. Ingresar una cadena de caracteres, en caso de que la misma se corresponda con un **entero**, realizar la conversión al formato número **utilizando `atoi()`**. Presentar por pantalla la cadena ingresada y el número convertido. Tener en cuenta que para representar un entero debe poseer solamente caracteres numéricos o el signo menos al principio. Ejemplo: Se ingresa 7291 -> Muestra la cadena 7291 y el número 7291.
9. Ingresar una cadena de caracteres, en caso de que la misma se corresponda con un número

decimal positivo, es decir que posea solamente caracteres numéricos y el punto decimal, realizar la conversión al formato número. Usar **atof()**. Presentar por pantalla la cadena ingresada y el número convertido.

10. Leer una cadena de caracteres que sólo puede contener dígitos y barra, con el formato dd/mm/aaaa. Mostrar por pantalla un mensaje indicando si la cadena ingresada es una **fecha válida**, es decir:
 - a. solo puede contener números y barra
 - b. los valores de día, mes y año se encuentran en los rangos permitidos:
 - el día debe estar entre 1 y 30 (para todos los meses).
 - el mes entre 1 y 12 .
 - el año debe ser menor o igual a 2023.

Por ejemplo: 34/05/2018, no es válida porque no hay meses de 34 días.

Sugerencia: investigar qué función de string.h puede servir para esto

11. Ingresar dos cadenas de caracteres (apellidos y nombres). Convertir el primer carácter de cada una a mayúscula. Recordar que una persona puede tener más de un apellido o nombre. Generar una nueva cadena concatenando las dos anteriores. **Usar strcat()**.
12. Escribir un programa que solicite al usuario una cadena de caracteres y un número entero positivo(n). Luego, cifre la cadena utilizando el "Cifrado César" desplazando cada letra n posiciones hacia la derecha en el alfabeto. Por ejemplo, si n=2, "HOLA" se cifraría como "JQNC".

Arreglo de estructuras que contienen variables simples y cadenas

Para los siguientes problemas usar la siguiente estructura:

```
typedef struct{
    char apellido[40];
    char nombre[40];
    char ciudad[20];
    float precio;
    int dias;
}huesped;
```

13. Ingresar los datos de **UN** huésped. Luego presentar por pantalla los datos del mismo (apellido/s, nombre/s, ciudad de origen, cantidad de días que se va a hospedar, importe que debe abonar por los días que se quede). Para presentar por pantalla usar este formato:

Huesped: Younes, José	Ciudad de origen: Cordoba
Dias Hospedaje: 5 dias	Importe a abonar: 7500.00
14. Agregar los siguientes controles al problema anterior
 - a. Controlar que los valores numéricos cargados sean válidos.
 - b. Los campos apellido/s, nombre/s solo deben ser alfabéticos, y pueden contener caracteres especiales como apóstrofe y espacio.

15. Ingresar los datos de **N** huéspedes de un Hotel, luego mostrarlos por pantalla:

Huesped	Ciudad de origen
Younes, José	Cordoba
Juarez, Julio José	Misiones
Rodriguez, María	Cordoba

16. Ingresar los datos de **N** huéspedes de un Hotel, luego mostrarlos por pantalla ordenados alfabéticamente por su apellido.
17. Ingresar los datos de **N** huéspedes de un Hotel, y solicitar una ciudad. Luego mostrar por pantalla solamente a las personas que provengan de esa ciudad. Usar strcmp. En caso de no haber huéspedes de esa ciudad mostrar un mensaje indicando esa situación.
18. Escriba un programa en lenguaje C que permita gestionar una lista de huéspedes de un Hotel. El programa debe poseer un menú con las siguientes opciones:

- ☐ Agregar un huésped a la lista. Consejo: usar lo realizado en el punto 15.
- ☐ Mostrar por pantalla la lista de huéspedes ordenada alfabéticamente. Consejo: usar lo realizado en el punto 7 y 16.
- ☐ Buscar huéspedes por apellido, en caso de no encontrar ninguno mostrar un mensaje.
- ☐ Dada una ciudad, indicar la cantidad de huéspedes que son de esa ciudad. Consejo: puede servir lo realizado en el ejercicio 17.
- ☐ Salir del programa.

19. Ingresar los datos de **N** fechas en un arreglo de estructuras y mostrar el vector de fechas resultantes en el main. Usar la siguiente estructura:

```
struct fecha {
    int dia, mes, anio;
};
```

Realizar una función **fecha leerFecha()** ; que deberá :

- i. realizar el ingreso de datos como caracteres,
- ii. verificar que cada carácter sea un entero (*) y
- iii. realizar la conversión a entero
- iv. si se convierte correctamente validar el rango de valores:
 1. día debe estar entre 1 y 30 (para todos los meses),
 2. el mes entre 1 y 12
 3. y el año debe ser menor o igual al año actual, omitir consideraciones como años bisiestos, etc

Luego de corroborar esto, deberá dentro de esta misma función llamar a

```
int control Fecha(fecha una Fecha);
```

Que se encargará de verificar si la fecha leída es menor a la fecha actual. En caso de serlo devolverá 1, en caso contrario -1. Si retorna -1 dentro de leerFecha deberá volver a solicitar el ingreso de la fecha. Para la fecha actual cargarla en el programa como un valor fijo.

() Por ejemplo si por teclado se ingresa día = "ab" se deberá volver a solicitar la carga del valor del día, ahora bien si se ingresa día = "23" se procederá a solicitar el mes realizar el mismo control y lo mismo con el año.*

Ejercicios combinados

20. Ingresar N cadenas de caracteres. Luego ingresar otra cadena y buscar si dicha cadena se encuentra en el arreglo, presentar por pantalla la cantidad de veces que se encontró dicha cadena.
21. Ingresar N apellidos (tener en cuenta que cada persona puede tener uno o más apellidos), luego presentar por pantalla los apellidos ordenados en forma alfabética y con la primera letra de cada apellido en mayúsculas y el resto en minúsculas.
22. Ingresar N cadenas de caracteres. Luego imprimir en pantalla solamente las cadenas que comiencen con un carácter alfabético en mayúscula.
Ejemplo: se ingresan 6 cadenas de carácter: primer cuaderno; Segunda persona; computadora; Pre procesador; Apellido; silla lado pasillo. Debe mostrar las cadenas: Segunda persona; Pre procesador, Apellido.

Argumentos de main

23. Escribe un programa en C que reciba dos números enteros como argumentos en la función main y muestre la suma de los dos números por pantalla.
24. Ingresar por línea de comandos, dos palabras (realizar el control de la cantidad de palabras ingresadas). Luego mostrar por pantalla las palabras ordenadas en forma alfabética, o un mensaje en caso de que sean iguales.
25. Escribe un programa en C que reciba una cadena de caracteres como argumento en la función main y muestre la longitud de la cadena por pantalla.