

## EXPLICACIÓN EJERCICIOS

### SIMPLE C QUESTIONS :

#### 1. Area Of Triangle:

Para encontrar el área de un triángulo necesitamos la altura y la base.

```
Digite base y altura del triangulo  
-> 20 30  
  
El Area del triangulo es : 300.000000
```

#### 2. Factorial Of A Given Number:

El factorial de un entero no negativo  $n$ , denotado por  $n!$ , es el producto de todos los enteros positivos menores o iguales a  $n$ .

```
Ingresa un numero(entero): 20  
El factorial de 20 = 2432902008176640000
```

#### 3. Temperature Celsius into Fahrenheit:

Por medio de las ecuaciones de física, se calcula y convierte la temperatura a la que desea.

Fahrenheit =  $((9/5) * \text{Celsius}) + 32$ ;

Celsius =  $5 * (\text{Fahrenheit} - 32) / 9$ ;

```
Seleccione una opcion  
1. Fahrenheit a Celsius  
2. Celsius a Fahrenheit  
-> 1  
  
Digite el valor de la temperatura Fahrenheit: 20  
  
Temperatura Celsius: -6.666667
```

### IF/ELSE STATEMENT :

#### 4. Check Number Is Even Or Odd:

La lógica es muy simple, ya que sabemos que si el número es divisible por 2, entonces el número es par o impar.

```
Digite cualquier número para comprobar si es par o impar :  
24  
24 es PAR
```

#### 5. Leap Year:

El usuario debe introducir el año y el programa debe mostrar el mensaje "El año introducido es bisiesto o no". Los años bisiestos tienen un total de 366 días y un 29 de febrero. Si necesitamos encontrar qué año es bisiesto o no.

```
juan@Kaiceer: ~/Documentos
Digite un año: 2024
2024 es un año bisiesto.
juan@Kaiceer: ~/Documentos
```

#### 6. Find Greatest Among Three Numbers:

Para este problema hay una lógica simple que toma los tres números y escoge un número y compara ese número con el resto de los dos números, si el número es el mayor, entonces, imprime, si no, escoge el segundo número y repite el proceso anterior, si el mayor, entonces, imprime el segundo número es el mayor, si no, va al tercer número y repite el mismo proceso.

```
juan@Kaiceer: ~/Documentos
Digite 3 numeros: 10
50
20
50.00 es el numero mayor.
```

### LOOPS :

#### 7. Fibonacci:

Para imprimir la serie de Fibonacci utilizamos un método simple, ya que sabemos que la serie de Fibonacci comienza con cero (0) y el siguiente elemento es uno.

```
juan@Kaiceer: ~/Documentos
Digite el tamaño que quiere ver de la serie de Fibonacci :
10
Serie Fibonacci: 0 , 1 , 1 , 2 , 3 , 5 , 8 , 13 , 21 , 34 ,
```

#### 8. Reverse A Number:

La lógica para invertir un dígito es simple, tomemos un ejemplo 12345 ahora primero dividimos un número con 10 así que obtenemos el recordatorio 5 almacenamos el recordatorio y luego y usando esta lógica multiplicamos  $rev = rev * 10 + rem$  y de nuevo dividimos el número con 10, ahora el número es 1234 y repetimos el proceso una y otra vez hasta que el número sea mayor o igual a 1.

```
juan@Kaiceer: ~/Documentos/Arquitectura/2doCorte
Digite cualquier número que desee invertir :
2983
Numero Invertido : 3892
juan@Kaiceer: ~/Documentos/Arquitectura/2doCorte
```

#### 9. Count Number of Digits of a Number (Integer):

Cuenta el número de dígitos de un número (entero) usando el bucle While.

```
Digite cualquier numero para contar digitos :  
12345  
El numero de digitos es = 5
```

## SWITCH CASE :

### 10. Convert Celsius To Fahrenheit And Vice Versa Using Switch:

Para convertir la temperatura de Celsius a Fahrenheit, utilizamos una fórmula determinada. Sólo tenemos que tomar un valor o la temperatura en grados Celsius o Fahrenheit por el usuario y poner los valores en la fórmula dada e imprimir el resultado dado por la fórmula en la pantalla.

```
Seleccione una opcion  
1. Fahrenheit a Celsius  
2. Celsius a Fahrenheit  
-> 1  
  
Digite el valor de la temperatura Fahrenheit: 20  
  
Temperatura Celsius: -6.666667
```

### 11. Calculator:

programa de calculadora simple usando un switch, para cada tipo de operación.

```
Digite el primer valor:20  
Digite un operador(+, -, *, /):+  
Digite el segundo valor:20  
Resultado es = 40
```

### 12. Find Radius, Circumference and Volume of Cylinder:

Como podemos ver, sólo tenemos que aplicar la fórmula del área del círculo o la circunferencia del círculo y el volumen del cilindro e imprimir la salida en la pantalla, pero también tenemos que poner la condición en el caso del interruptor para que los usuarios puedan elegir fácilmente su elección y realizar una operación adecuada. En este problema el valor de PI es fijo hasta 10 números decimales  $PI = 3.1415926535$ .

**Area of Circle** =  $PI * Radius * Radius$

**Circumference of Circle** =  $2 * PI * Radius$

**Volume of Cylinder** =  $PI * Hight * Radius * Radius$

```

=====
Que operacion quiere calcular?
1. Area del Circulo
2. Circunferencia del Circulo
3. Volumen del Cilindro
0. Salir
=====

Digite la opcion que desea:1

Digite el radio del Circulo:28

=====

Area del Circulo = 2462.98755
=====

```

## ARRAY QUESTIONS :

### 13. Print "I AM IDIOT" Instead Of Your Name Using Array

Ejercicio en el cual se le pide el nombre al usuario y este al reconocer un carácter lo intercambia por cada letra de la palabra "I AM IDIOT"

```
Escribe tu nombre -> I AM AN IDIOT
```

### 14. Convert All Input String Simultaneously Into Asterisk ( \* )

Ejercicio en el cual se pide una entrada de texto y cada vez que el programa reconoce un carácter este lo cambia por un asterisco(\*)

```
*****
```

### 15. Delete Element From Array At Desired Or Specific Position

Ejercicio que pide introducir elementos a una matriz para después pedir al usuario que elija la posición que desea eliminar de dicha matriz

```

Introduzca el numero de elementos de la matriz -> 3

Introduzca 3 elementos para la matriz ->
1
2
3

Ingresa la posicion del elemento a eliminar -> 1

Matriz despues de borrar un elemento : 2 3

```

## MATRIX QUESTIONS :

### 16. Add two matrices

Ejercicio en el cual se le pide al usuario que digite dos matrices para posteriormente poder realizar la suma de estas 2 matrices

```

Introduzca los elementos de la primera matriz :
Ingrese el elemnto a11: 2
Ingrese el elemnto a12: 2
Ingrese el elemnto a21: 2
Ingrese el elemnto a22: 2

Introduzca los elementos de la segunda matriz :
Ingrese el elemnto a11: 2
Ingrese el elemnto a12: 2
Ingrese el elemnto a21: 2
Ingrese el elemnto a22: 2

Suma de las 2 matrices :
4  4
4  4

```

### 17. Subtract two matrices

Ejercicio en el cual se le pide al usuario que digite dos matrices para posteriormente poder realizar la resta de estas 2 matrices

```

Introduzca los elementos de la primera matriz :
Ingrese el elemnto a11: 4
Ingrese el elemnto a12: 4
Ingrese el elemnto a21: 4
Ingrese el elemnto a22: 4

Introduzca los elementos de la segunda matriz :
Ingrese el elemnto a11: 2
Ingrese el elemnto a12: 2
Ingrese el elemnto a21: 2
Ingrese el elemnto a22: 2

Resta de las 2 matrices :
2  2
2  2

```

### 18. Check whether two matrices are equal or not

Ejercicio en el cual se le pide al usuario que digite dos matrices para posteriormente poder revisar si estas dos matrices son iguales

```

Introduzca los elementos de la matriz 1 :
2
2
2
2
2

Introduzca los elementos de la matriz 2 :
1
1
1
1

Matriz A no es igual a la matriz B

```

## STRING QUESTIONS LIST :

### 19. A String is Palindrome or Not

Ejercicio en el cual se le solicita a un usuario una frase y el programa revisará si esta frase es palíndromo. Es decir si se escribe igual de adelante hacia atrás y de atrás hacia adelante

## 20. Compare two strings

Ejercicio en el cual se le pide al usuario digitar 2 frases y el programa mediante la función strcmp() compara si estos dos strings son iguales

```
Ingrese el primer string -> hola
Ingrese el segundo string -> hola
Los string son iguales
```

## 21. Concatenate two strings

Ejercicio en el cual se le pide al usuario que ingrese 2 strings y el programa hace la respectiva unión de dichos strings

```
Strings concatenados : Hola Mundo!
```

## STRING QUESTIONS LEVEL UP :

### 22. Find the first occurrence of a character in a given string.

Ejercicio en el cual se le pide al usuario un string, posteriormente se le pide ingresar el carácter que desea buscar y el programa arroja un mensaje diciendo si se encontró o no el carácter y en qué posición se encuentra

```
Ingrese el string -> arquitectura
Introduzca el caracter a buscar -> r
Caracter 'r' encontrado en : 1
Caracter 'r' encontrado en : 10
```

### 23. Find the last occurrence of a character in a given string.

Ejercicio en el cual se le pide al usuario un string y el programa encuentra la última aparición del carácter en la cadena dada

```
Ingrese el string -> arquitectura
Introduzca el caracter a buscar -> i
Caracter 'i' encontrado en : 5
```

### 24. Find the highest frequency character in a string.

Ejercicio en el cual se pide al usuario que ingrese una cadena y el programa calcula el carácter de mayor frecuencia en dicha cadena

```
Ingrese un string -> arquitectura

Maximo caracter repetido en la cadena = a
Ocurre 2 veces
```

## FUNCTION QUESTIONS :

### 25. Convert Binary Number to Decimal or Decimal Number to Binary:

Se numeran los dígitos de derecha a izquierda comenzando desde cero, a cada número se le asigna la correspondiente potencia base 2 y al final se suman las potencias.

```
Instructions:
1. Enter alphabet 'd' to convert binary to decimal.
2. Enter alphabet 'b' to convert decimal to binary.
d
Enter a binary number: 110101001
110101001 in binary = 425 in decimal
-----
```

### 26. Display Prime Numbers Between Intervals:

Programa que determina todos los números primos que hay dentro de un intervalo.

```
Digitate dos numeros como intervalos: 100 200
Numeros primos entre 100 y 200 son: 101 103 107 109 113 127 131 137 139 149 151 157 163 167 173 179 181 191 193 197 199
```

### 27. Check Prime or Armstrong Number:

Programa que verifica si el número que se digita es un número primo o un número Armstrong, es decir, número que es igual a la suma de la n-ésima potencia de los dígitos

```
Enter a positive integer: 371
Enter P to check prime and A to check Armstrong number: a
371 is an Armstrong number.
```

## POINTER QUESTIONS :

### 28. Reverse a String Using Pointers:

Invertir una cadena usando punteros es muy probable que el programa de pila también siga la misma regla de pila FILO (First In Last Out) en Invertir una cadena usando punteros estamos haciendo lo mismo cuando estamos tomando la entrada del usuario al mismo tiempo estamos calculando la longitud de una cadena y con la ayuda de la longitud uno por uno estamos transfiriendo el carácter uno por uno de una cadena a otra y hasta que la longitud no se convierte en cero en el bucle while ponemos la misma condición.

```
Enter The String Without Space:
GhanendraPratapSinghYadav

Reverse String Is
ceYadavYhgnisPatarPardnenahG
-----
```

### 29. Compare Two String Using Pointer:

En este problema, estamos comparando dos cadenas usando punteros para comparar dos cadenas, no estamos usando la función strcmp, estamos comparando con la ayuda de punteros. Primero tomamos una entrada a través del usuario, primero tomamos una primera cadena y una segunda cadena y asignamos ambas cadenas en dos variables de tipo puntero por separado, después ejecutamos un bucle while y ponemos la condición de que la cadena 1 y la cadena 2 sean iguales y también que ambas cadenas no sean iguales a null hasta que se incremente la cadena en un carácter y se repita el proceso una y otra vez. Si la primera cadena es igual a la segunda cadena entonces imprime el mensaje "Ambas cadenas son iguales" si no entonces imprime "Ambas cadenas no son iguales" en la pantalla.

```
Enter The First String: ghanendra
Enter The Second String: ghanendra

Both Strings Are Equal.
-----
C:\Users\Ghanendra> gcc 344.c -o 344.exe -lm -std=c99
```

## FILE HANDLING :

### 30. Display its Own Source Code as Output:

Así que básicamente tenemos que tomar una entrada de un nombre de archivo como una fuente de nombre del programa con la extensión .c/cpp/.java y muchos otros y presionar el enter después de golpear el programa en sí se imprimirá en la pantalla de la consola o podemos decir que el archivo .c/cpp/.java contiene con estas extensiones se imprimirá en la pantalla. En otras palabras, podemos decir que el programa mismo será abierto.

```
Enter The File Name You Want To Open
file open itself.c

Your File If Given Below file open itself.c :

#include <stdio.h>
#include <stdlib.h>

int main()
{
    char ch, filename[25];
    FILE *fp;

    printf("Enter The File Name You Want To Open \n");
    gets(filename);

    fp = fopen(filename, "r");

    if( fp == NULL )
    {
        perror("Error While Opening The File. Or File Not Exist\n");
        exit(EXIT_FAILURE);
    }

    printf("\nYour File If Given Below %s :\n\n", filename);

    while( ( ch = fgetc(fp) ) != EOF )
        printf("%c", ch);

    fclose(fp);
    return 0;
}
```



## **SORTING :**

### **31. Bubble sort in C**

Programa en el cual se le pide al usuario que ingrese la cantidad de elementos que desea ordenar. Posteriormente se le pide ingresar los números enteros para así poder realizar el orden ascendente correspondiente

```
Introduzca el numero de elementos -> 5
Introducir 5 enteros :
2
4
8
1
0
Lista clasificada en orden ascendente :
0
1
2
4
8
```

### **32. Selection Sort**

Programa en el cual se le pide al usuario que ingrese la cantidad de elementos que desea ordenar. Posteriormente se le pide ingresar los números enteros para así poder realizar el orden ascendente correspondiente

```
Ingrese el numero de elementos -> 2
Ingrese 2 numeros :
1
2
Matriz Ordenada :
1
2
```

### **33. Shell Sort**

Programa en el cual se le pide al usuario que ingrese la cantidad de elementos que desea ordenar. Posteriormente se le pide ingresar los números enteros para así poder realizar el orden ascendente correspondiente

```
Sorted array:
1 3 4 5 6 7 8 9
```

## **SEARCHING :**

### **34. Binary Search in C**

Ejercicio en el cual se le pide al usuario ingresar números enteros, posteriormente se le pide ingresar el número que desea buscar y el programa arroja un mensaje diciendo si se encontró o no el carácter y en qué posición se encuentra

```
Ingrese el numero de elemntos ->
5
Ingrese 5 enteros :
1
2
3
4
5
Introduzca el valor a encontrar -> 5
5 Encontrado en el lugar 5.
```

### 35. Binary Search in C Recursive

Ejercicio en el cual se le pide al usuario ingresar números enteros, posteriormente se le pide ingresar el número que desea buscar y el programa de forma recursiva arroja un mensaje diciendo si se encontró o no el carácter y en qué posición se encuentra

```
Element is present at index 3
=====
```