**EJERCICIOS DE PROGRAMACION PARA EL SEGUNDO PARCIAL**

1.      Que pida un número y diga si es primo o no.

2.      Que muestre los números del 1 al 100.

3.      Que muestre los números del 100 al 1.

4.      Que muestre los números pares que haya del 1 al 100.

5.      Que muestre los números impares que haya del 1 al 100.

6.      Que imprima la suma de todos los números que van del 1 al 100.

7.      Que imprima la suma de todos los números pares que van del 1 al 100.

8.      Que imprima la suma de todos los números impares que van del 1 al 100.

9.      Que imprima la suma de todos los números pares que van del 1 al 100 y diga cuántos hay.

10.  Que imprima la suma de todos los números impares que van del 1 al 100 y diga cuántos hay.

11.  Que pida dos números y muestre todos los números que van desde el primero al segundo. Se debe controlar que los valores son correctos.

12.  Que pida dos números y muestre todos los números pares que van desde el primero al segundo. Se debe controlar que los valores son correctos.

13.  Que pida dos números y muestre todos los números impares que van desde el primero al segundo. Se debe controlar que los valores son correctos.

14.  Que pida dos números y sume todos los números que van desde el primero al segundo. Se debe controlar que los valores son correctos.

15.  Que pida dos números y multiplique todos los números que van desde el primero al segundo. Se debe controlar que los valores son correctos.

16.  Que pida un número y muestre en pantalla el mismo número de asteriscos.

17.  Que muestre los números del 1 al 100 en una tabla de 10x10.

18.  Que escriba las tablas de multiplicar del 0 al 10.

19.  Que muestre la tabla de multiplicar de un número cualquiera.

20.  Que calcule la media de 10 números.

23.  Que calcule la media de X números, se dejarán de solicitar números hasta que se introduzca el cero.

24.  Que calcule la suma de los cuadrados de los 100 primeros números.

25.  Que escriba los primeros 25 dígitos de la sucesión de Fibonacci. La serie de Fibonacci es la sucesión de números:  
0, 1, 1, 2, 3, 5, 8, 13, 21, 34, ...  
  
Cada número se calcula sumando los dos anteriores a él.

26. Hacer un **programa en C++** que lea n números y determine cuál es el mayor, el menor y la media de los números leídos.

27. Hacer un **programa en C++** que permita sumar la sucesión de los números n,

siendo n un número que se ingresa por medio del teclado.

28. Escribir un programa que calcule la suma de los cuadrados de los 100 primeros números enteros.

29. **Escribir un programa que lea 10 datos desde el teclado y sume sólo aquellos que sean negativos.**

**30.** Escriba un programa que calcule el valor de ex Recuerde que dicha constante matemática se calcula de la siguiente forma:

31.- Considere siguiente proceso repetitivo para un número entero dado: si el número es 1, el proceso termina. De lo contrario, si es par se divide entre 2, y si es impar se multiplica por 3 y se le suma 1. Si empezamos con 6, por ejemplo, obtendremos la sucesión de números 6, 3, 10, 5, 16, 8, 4, 2, 1. La conjetura de Collatz dice que, a partir de cualquier número inicial, la sucesión obtenida siempre termina en 1. Escriba un programa que permita verificar la conjetura de Collatz para cualquier entero dado, y que imprima la secuencia correspondiente.

32. El juego de dados conocido como “craps” (tiro perdedor) es muy popular, realice un programa que simule dicho juego, a continuación se muestran las reglas para los jugadores. • Un jugador tira dos dados. Cada dato tiene seis caras. Las caras contienen 1, 2, 3, 4, 5 y 6 puntos. • Una vez que los dados se hayan detenido, se calcula la suma de los puntos en las dos caras superiores. • Si a la primera tirada, la suma es 7, o bien 11, el jugador gana. • Si a la primera tirada la suma es 2, 3 o 12 (conocido como “craps”), el jugador pierde (es decir la casa “gana”). • Si a la primera tirada la suma es 4, 5, 6, 8, 9 ó 10, entonces dicha suma se convierte en el “punto” o en la “tirada”. • Para ganar, el jugador deberá continuar tirando los dados hasta que haga su “tirada”. • El jugador perderá si antes de hacer su tirada sale una tirada de 7.

33. Un número entero es un “número perfecto” si sus factores, incluyendo al 1 (pero excluyendo en el número mismo), suman igual que el número. Ejemplo: 6 es un numero perfecto porque 6= 1+2+3. Escriba una función que regrese los primeros 100 números perfectos. Esta función debe tener una función anidada que determine al número perfecto.

34. Escriba una función que tome un valor entero de cuatro dígitos y regrese el número con los dígitos invertidos. Por ejemplo, dado el número 7631, la función deberá regresar 1367.

1.-Escribe un programa que reciba un número entero N del usuario e imprima una línea Horizontal de N

asteriscos.

35.- Escribe un programa que reciba un número entero N del usuario e imprima una línea vertical de N

asteriscos.

36.- Escribe un programa que reciba un número entero N del usuario e imprima un cuadrado de asteriscos

de N x N.

37.- Escribe un programa que reciba dos números enteros N y M del usuario e imprima un rectángulo de 2

asteriscos de N x M.

38.- 10. Escribe un programa que reciba un número entero N del usuario e imprima el contorno de un

cuadrado de N x N de asteriscos. Ej.

>> 3

\*\*\*

\* \*

\*\*\*

39. Escribe un programa que reciba dos números enteros N y M del usuario e imprima el contorno

de un rectángulo de N x M de asteriscos. Ej.

>> 3 5

\*\*\*\*\*

\* \*

\*\*\*\*\*

40. Escribe un programa que reciba un número entero N del usuario e imprima una escalinata de

N pisos de asteriscos. Ej.

>> 4

\*

\*\*

\*\*\*

\*\*\*\*

41. Escribe un programa que reciba un número entero N del usuario e imprima una escalinata de

N pisos de asteriscos. Ej.

>> 4

\*

\*\*

\*\*\*

\*\*\*\*

42. Escribe un programa que reciba un número entero N del usuario e imprima una escalinata

invertida de N pisos de asteriscos. Ej.

>> 4

\*\*\*\*

\*\*\*

\*\*

\*

43. Escribe un programa que reciba un número entero N del usuario e imprima una escalinata 5

invertida de N pisos de asteriscos. Ej.

>> 4

\*\*\*\*

\*\*\*

\*\*

\*

44. Escribe un programa que reciba un número entero N del usuario e imprima una diagonal

invertida de N asteriscos. Ej.

>> 4

\*

\*

\*

\*

45. Escribe un programa que reciba un número entero N del usuario e imprima una diagonal de N

asteriscos. Ej.

>> 4

\*

\*

\*

\*

46. Escribe un programa que reciba un número entero N del usuario e imprima una pirámide de N

pisos de asteriscos. Ej.

>> 4

\*

\*\*\*

\*\*\*\*\*

\*\*\*\*\*\*\*

47. Escribe un programa que reciba un número entero N del usuario e imprima una pirámide

invertida de N pisos de asteriscos. Ej.

>> 4

\*\*\*\*\*\*\*

\*\*\*\*\*

\*\*\*

\*

48. Escribe un programa que reciba un número entero N del usuario e imprima el contorno de

pirámide de N pisos de asteriscos. Ej.

>> 3

\*

\* \*

\*\*\*\*\*

49. Escribe un programa que reciba un número entero N del usuario e imprima un rombo de N

pisos de asteriscos. El número debe ser mayor que tres e impar para dibujar el rombo. En caso

contrario se debe mostrar un mensaje que indique las restricciones y terminar. Ej.

>> 5

\*

\*\*\*

\*\*\*\*\*

\*\*\*

\*

50. Escribe un programa que reciba un número entero N del usuario e imprima una pirámide 6

horizontal de N pisos de asteriscos. El número N debe ser impar para imprimir la pirámide. Ej.

>> 5

\*

\*\*

\*\*\*

\*\*\*

\*

51. Escribe un programa que reciba un número entero N del usuario e imprima una pirámide

horizontal de N pisos de asteriscos. El número N debe ser impar para imprimir la pirámide. Ej.

>> 5

\*

\*\*

\*\*\*

\*\*

\*

52. Escribe un programa que reciba un número entero N del usuario e imprima un reloj de arena

de N pisos de asteriscos. El número N debe ser impar para imprimir el reloj. Ej.

>> 5

\*\*\*\*\*

\*\*\*

\*

\*\*\*

\*\*\*\*\*

53. Escribe un programa que reciba un número entero N del usuario e imprima una X de N pisos

de asteriscos. El número N debe ser impar para imprimir la X. Ej.

>> 5

\* \*

\* \*

\*

\* \*

\* \*

54. Escriba un programa que despliegue el siguiente patrón.

\* \* \* \* \*

\* \* \* \* \*

\* \* \* \* \*

\* \* \* \* \*

55. Escriba un programa que lea un entero y determine e imprime cuantos dígitos de ese entero

son 7s.

56. Diseñar un algoritmo para imprimir la suma de los cuadrados de los 100 primero enteros

57. En matemáticas, la siguiente expresión es siempre verdadera:

Diseñar un algoritmo para verificar esta propiedad para valores que van de -10.00 a +10.00 en pasos de 0.2

58. Los pescadores profesionales deben informar mensualmente la cantidad pescada al Departamento de Pesca. Estos datos se analizan regularmente para determinar el crecimiento o la disminución de la cantidad total de algunas especies de peces y para indicar cualquier posible problema . Hacer un algoritmo que lea los siguientes datos para cada pescador:

Región de pesca ( en código numérico; 1,2,3 ó 4)

Número de la especie ( en código númérico del 1 - 5)

Número pescado en este año ( entero)

Numero pescado en el año anterior ( entero)

Y calcule un crecimiento o disminución desproporcionados mediante la siguiente fórmula:

Si hay un porcentaje que supere el 30% indicarlo con un mensaje

59.- Un distribuidor de juguetes ha hecho una excelente compra de 10 000 juguetes en cajas rectangulares de diversos tamaños. El distribuidor desea poner las cajas en esferas plásticas de brillantes colores y sellarlas como paquetes sorpresa. Las esferas son de cuatro diámetros diferentes: 4,6,8 y 10 pulgadas, por lo que para realizar todo en orden desea saber cuántas esferas de cada diámetro debe comprar. Puesto que la diagonal de una caja rectangular cuyas dimensiones son A, B, y C, está dada por :

Esto es, además la dimensión mayor, el distribuidor debe calcular las longitudes de las diagonales de las cajas y determinar el número de las que son de 4 pulgas o menos, las comprendidas entre 4 y 6 pulgas, etc. Las dimensiones e cada caja se leen por separado. Diseñar un algoritmo para leer estos datos y determinar el número de esferas de cada tamaño que se necesitan para empacar los juguetes.

60. La oficina gubernamental de seguros de Saskatchewan ha reunido datos concernientes a todos los accidentes de tránsito ocurridos en la provincia en el último año. Para cada conductor involucrado en un accidentes se necesitan los siguientes datos:

Año de nacimiento (entero)

Sexo (M =1, F= 2)

Código de registro (1 para los registrados en Saskatchewan, 0 para los demás).

Diseñe un algoritmo para leer todos los datos e imprimir los siguientes resúmenes estadísticos de los conductores que se han visto envueltos en accidentes:

1. Porcentaje de conductores menores de 25 años.
2. Porcentaje de conductores del sexo femenino
3. Porcentaje de conductores del sexo masculino con edades comprendidas entre 18 y 15 años
4. Porcentaje de conductores con registros fuera de la provincia.

Usar el método de fin de archivo para detectar el término de los datos de entrada.

61. Diseñar un algoritmo para calcular y tabular los valores de la función:

Para x= 2,4,6,8. y = 6,9,12,15,18,21

62. Diseñar un algoritmo para calcular el número de puntos con coordenadas de valores enteros que están contenidos en la siguiente elipse:

63. La criba de Eratóstenes, renombrado astrónomo y geógrafo griego del s. III a.C. es una técnica para generar números primos. Se comienza escribiendo todos los enteros impares desde 3 hasta N; en seguida se elimina cada tercer elemento después de 3, cada quinto elemento después de 5, etc., hasta que los múltiplos de todos los enteros impares menores que hayan sido eliminados. Los enteros que quedan constituyen la lista exacta de los números primos entre 3 y N. Diséñese un algoritmo para generar los números primos entre 3 y N utilizando la técnica de la criba.