EJERCICIOS DE ESTRUCTURAS DE REPETICIÓN.

Hacer los siguientes algoritmos/programas:

1. De la lista de problemas que están al final del capítulo 2 del libro, resolver los ejercicios 14,16, 17, 18, 20,22, 23, 23, 26, 30,31.
2. De la lista de problemas que están al final del capítulo 4 del libro, resolver los ejercicios 10, 13,14, 18.
3. Hacer un algoritmo que genere y muestre 8 valores aleatorios impares mayores que N, y muestre su suma. N es ingresado por el usuario.
4. Hacer un algoritmo que genere y muestre 7 números aleatorios en el rango de 5 al 100. Mostrar la suma de ellos así como el mayor y el menor generados.
5. Hacer un algoritmo que imprima **N** términos de una suma que comienza con un número ***inicial*** indicados por el usuario. Ejemplo: Si el número **inicial** es 20 y **N** vale 3, entonces debe imprimirse la suma: **20+21+22= 63**. Al final del programa la suma debe verse tal como se mostró (En cada vuelta de ciclo se va imprimiendo un término).
6. En una empresa se desea calcular la productividad de un año, de acuerdo a lo siguiente:

   Enero, Febrero y Marzo tienen factor 15
   Abril, Mayo y Junio factor 17
   Julio y Agosto factor 19
   Septiembre, Octubre y Noviembre factor 20
   Diciembre factor 21

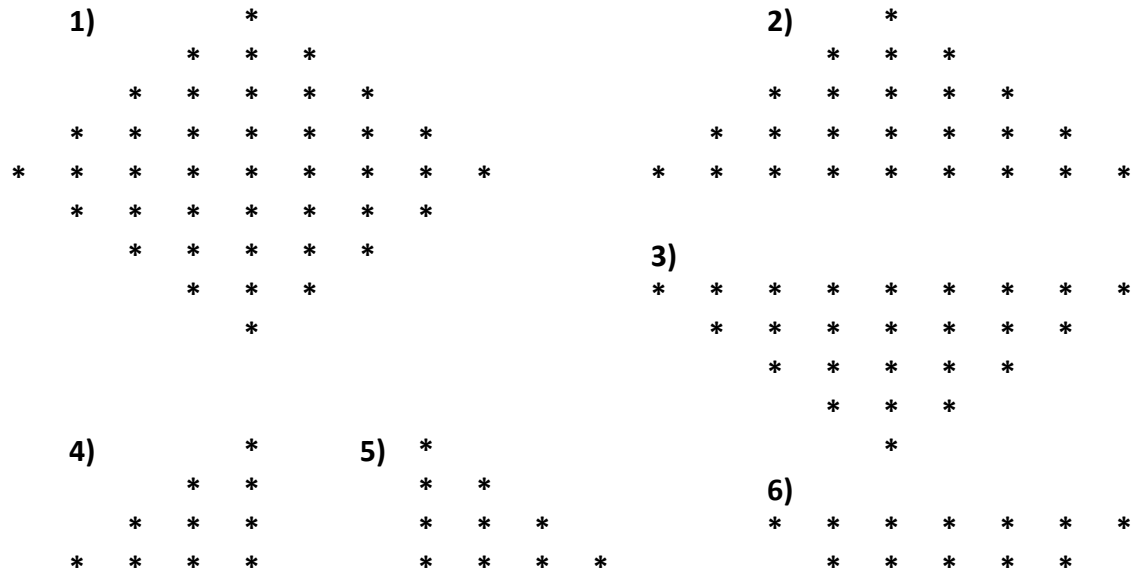
   Escribir un algoritmo que permita calcular la productividad ***anual***, si se sabe que ésta se calcula sumando los productos del número de artículos producidos *en cada mes*, multiplicado por el factor que le corresponde.

7. Hacer un algoritmo que solicite un número entero e imprima una a una las cifras que lo componen, comenzando por la última de ellas. Por ejemplo si el número dado fue 14009, el algoritmo debe imprimir **9, 0, 0, 4, 1**
8. Hacer un algoritmo que solicite un número entero e imprima una a una las cifras que lo componen, comenzando por la primera de ellas. Por ejemplo si el número dado fue 60809, el algoritmo debe imprimir **6, 0, 8,0, 9**
9. Hacer un algoritmo que solicite **N** números y después de ellos se muestre cuál fue el **mayor**, cuál fue el **menor** y el **promedio** de todos. Los ceros ingresados se ignoran (Ni siquiera forman parte del promedio). **N** es dado por el usuario, así como cada número.
10. Forzar al usuario a ingresar dos valores mayores que cero, usando ciclos. Mostrar el mínimo común divisor ***(que no sea 1)*** y el máximo común divisor de los dos números dados por el usuario. Por ejemplo, si consideramos los números 18 y 6, el mínimo común divisor de ellos es 2 y el máximo común divisor es 6. Si consideramos el 7 y el 21, el mínimo común divisor de ellos es 7 y el máximo común divisor es también 7. Si consideramos el 13 y el 7, no existe un mínimo común divisor de ellos, ni un máximo común divisor.
11. Hacer un algoritmo que solicite un número e imprima la leyenda **YES!** si la suma de sus cifras (dígitos) pares es igual a la suma de sus cifras impares. Por ejemplo para el número 1430 se imprimiría **YES!**
12. Hacer un algoritmo que calcule y muestre el resultado de multiplicar **N** X **M**, a través de *sumas sucesivas*, con el menor número de sumas posibles. Considere que **N** y **M** son enteros ingresados por el usuario y siempre recibirán valores mínimos de cero. (prohibido utilizar el operador *)
13. Hacer un algoritmo que calcule el resultado de dividir **N** / **M**, a través de restas sucesivas. Considere que **N** y **M** son valores ingresados por el usuario y siempre recibirán valores mínimos de cero. Mostrar cociente y residuo de la división (prohibido utilizar los operadores **/** y **mod**). Considere como error la división entre cero enviando un mensaje que así lo indique.
14. Hacer un algoritmo que calcule e imprima el factorial de **N**. **N** debe ser ingresado por el usuario y siempre será un valor mínimo de 0. El factorial de un número resulta de multiplicar ese número por los valores inferiores a él, hasta llegar a 1. Por ejemplo el factorial de 7 es el resultado de 7x6x5x4x3x2x1. Por definición matemática el factorial de 0 es 1.
15. Hacer un algoritmo que imprima la siguiente sucesión de números: $0, 1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 144, 233, 377 \ldots$

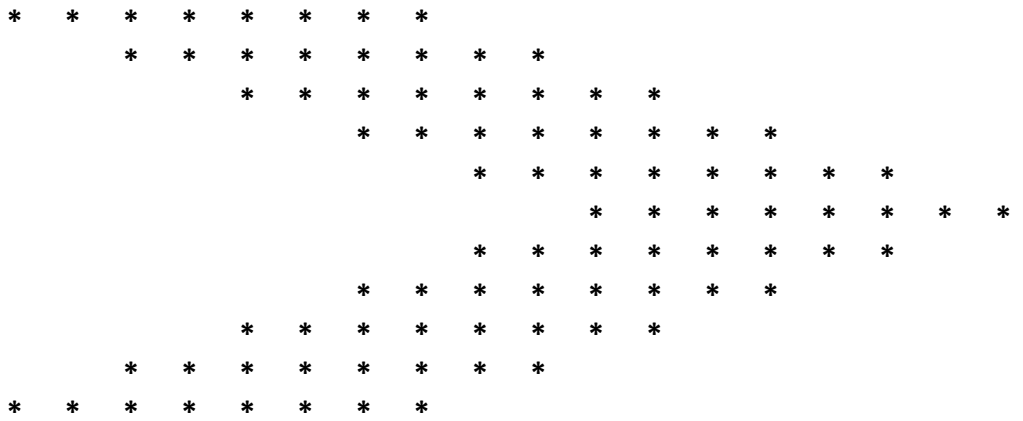
Observe que la sucesión comienza con los números 0 y 1, y a partir de éstos, los siguientes números se forman con la suma de los dos anteriores. El usuario debe ingresar la cantidad de valores que quiere ver, por ejemplo si el usuario indica que quiere ver 5 números, solo se imprimiría la sucesión 0,1,1,2,3

16. Hacer un algoritmo que eleve un numero **B** a una potencia **P. B** y **P** son ingresados por el usuario y deben ser valores positivos (validar a través de ciclos). Calcular el resultado con multiplicaciones sucesivas del valor B. Por ejemplo, si **B** vale 4 y **P** vale 6, entonces el resultado estará dado por multiplicar **P** veces el **B** (4x4x4x4x4x4)
17. Hacer un programa que solicite indefinidamente números enteros y al detenerse imprima cual la diferencia mayor entre dos números dados consecutivamente (uno en seguida del otro), así como los valores que generaron la diferencia. El programa deja de pedir números cuando se ingrese un cero. Por ejemplo, si se dieron los números 2, -10, 5, 50, -4, 0, el mensaje a imprimir sería: ***la diferencia mayor fue de 54 y se dio entre los valores 50 y -4***
18. (hasta ver java) Hacer un programa que genere e imprima aleatoriamente 10 números enteros entre 60 y 500. De esos 10 números se conservarán **el número mayor** y **el número menor** de los generados. A continuación se debe hacer una división del mayor entre el menor, **a través de restas sucesivas** (no utilizar el operador **/**, ni el **mod**), es decir, restar sucesivamente el menor al mayor. Finalmente mostrar el cociente y el residuo resultantes.
19. Imprimir en el rango de **N** a **M**, los números que cumplan con la condición de que la suma de sus cifras (dígitos) pares sea igual a la suma de sus cifras nones. **N** y **M** son dados por el usuario.
20. Hacer un algoritmo que solicite un número entero y lo almacene en otra variable pero al revés, finalmente imprimir el valor de ambas variables. Por ejemplo si el número dado fue 14009, el programa debe imprimir V**alor original: 14009, valor invertido: 90041**
21. Hacer un programa que solicite el ingreso de un valor entero **X** y muestre si ese número es caprichoso en la potencia de **P**. **P** también será ingresado por el usuario. Se dice que un número es caprichoso si la suma de sus dígitos elevada a cierta potencia es igual al mismo número. Por ejemplo, el número 4,913 es caprichoso en la potencia de 3, ya que $(4+9+1+3)^3$ resulta 4,913.
22. Hacer un programa que nos indique si dos números dados desde el teclado son amigos. Dos números amigos son dos enteros positivos *a* y *b* tales que *a* es la suma de los divisores propios de *b* y *b* es la suma de los divisores propios de *a*. (la unidad se considera divisor propio, pero no lo es el mismo número). Ejemplo: los valores 220 y 284 son amigos, ya que:
    - Los divisores propios de 220 son 1, 2, 4, 5, 10, 11, 20, 22, 44, 55 y 110, que suman 284.
    - Los divisores propios de 284 son 1, 2, 4, 71 y 142, que suman 220.

23. (Hasta ver tema de cadenas y caracteres en java). Hacer un programa con lo siguiente: un menú con el nombre de figuras de donde se pueda elegir la que se desea ver en ese momento. El usuario indicará en un inicio el caracter con el que desea ver la figura. Dentro del menú debe haber una opción para **cambiar el caracter** cuando se deseé. Después de imprimir una figura, volver al menú. El programa continua hasta que se elija la opción **terminar** del menú. Utiliza **'\n'** para imprimir el caracter en renglones sucesivos.



7)



24. (Hasta ver tema de cadenas) Hacer un programa que pregunte una cadena al usuario y muestre cuántas vocales hay en ella, (tanto mayúsculas como minúsculas). Por ejemplo, si la cadena introducida es: "Yo tenía diez PERRITOS", el programa deberá mostrar *La cadena dada tiene 9 vocales*.
25. (Hasta ver tema de cadenas) Hacer un programa que pida una cadena. Escribir un programa que valide que la cadena dada contenga únicamente los caracteres '0', '1', '2', '3', '4', '5', '6', '7', '8' y '9' y si es así transformar la cadena a un dato entero y elevarla al cuadrado mostrando el resultado. Ejemplo, si la cadena dada fue "167", entonces el programa imprimirá **167 al cuadrado es 27889**. Si la cadena dada fue "56h8" entonces el programa imprimirá ***Error la cadena no tiene formato de número***.
26. Solicitar el ingreso de una cadena y en otra poner los caracteres de la cadena ingresada, pero en reversa y en mayúsculas. Por ejemplo, si la cadena ingresada fue Mochila, entonces el mensaje de salida deben ser: ALIHCOM. La cadena ingresada podría incluir espacios.
27. Solicitar el ingreso de una frase y mostrar la misma frase pero con la inicial de cada palabra en mayúscula y el resto en minúsculas. Por ejemplo, si la frase ingresada fue: **Karen ESQuivEl murilLo**, entonces la frase a mostrar es: **Karen Esquivel Murillo**. Observe que en la segunda frase se deja solo un espacio entre cada palabra, aun cuando en la original había más de un espacio entre las segunda y tercera palabras.
28. Solicitar el ingreso de una cadena que contendrá una expresión aritmética que puede ser *suma, resta, división, multiplicación, división o módulo*. El programa debe separar la cadena en dos operandos y el operador correspondientes. Entonces debe hacerse la operación correspondiente y mostrar el resultado.
29. Solicitar dos cadenas y determinar si la segunda de ellas está contenida de manera exacta en la primera. Por ejemplo, si la primera cadena es **pollo pollito pollotes,** y la segunda cadena es **polla** entonces debe mostrarse el mensaje *"polla NO está contenida en la cadena pollo pollito pollotes"*. Si la segunda cadena es **polle** entonces debe mostrarse el mensaje *"polle NO está contenida en la cadena pollo pollito pollotes"*. Si la segunda cadena es **pollos** entonces debe mostrarse el mensaje *"pollos NO está contenida en la cadena pollo pollito pollotes"*. Si la segunda cadena es **ito** entonces debe mostrarse el mensaje *"ito SI está contenida en la cadena pollo pollito pollotes"*.