Ejercicios para practicar

# ESTRUCTURAS SECUENCIALES

1) Calcula el precio de un boleto de viaje, tomando en cuenta el número de kilómetros que se van a recorrer, siendo el precio $45 por Km.

2) Calcula el precio de un boleto de viaje, tomando en cuenta el número de kilómetros que se van a recorrer, siendo el precio $.10,50 por Km.

3) En un hospital existen 3 áreas: Urgencias, Pediatría y Traumatología. El presupuesto anual del hospital se reparte de la siguiente manera:

| Área | Presupuesto |
| --- | --- |
| Urgencias | 37% |
| Pediatría | 42% |
| Traumatología | 21% |

Obtener la cantidad de dinero que recibirá cada área para cualquier monto presupuestal.

4) Escriba un algoritmo que dado el número de horas trabajadas por un empleado y el sueldo por hora, calcule el sueldo total de ese empleado. Tenga en cuenta que las horas extras se pagan el doble. Las horas extras son aquellas que van a partir de las 40 horas.

5) Leer dos números y encontrar:

a. La suma del doble del primero más el cuadrado del segundo.

b. El promedio de sus cubos.

6) Leer tres números enteros de un Digito y almacenarlos en una sola variable que contenga a esos tres dígitos Por ejemplo si A=5 y B=6 y C=2 entonces X=562.

7) Un profesor desea saber qué porcentaje de hombres y qué porcentaje de mujeres hay en un grupo de estudiantes.

8) Elabore un programa que realice la conversión de cm. a pulgadas. a. Donde 1cm = 0.39737 pulgadas.

9) Se desea conocer el promedio y porcentaje de hombres y mujeres cursando el Programación I en UTN FRRe, teniendo como dato la cantidad de hombres y mujeres de 3 comisiones.

# ESTRUCTURAS CONDICIONALES

10) Determinar si un alumno aprueba o desaprueba un curso, sabiendo que aprueba si su promedio de tres calificaciones es mayor o igual a 60; desaprueba en caso contrario.

11) En un almacén se hace un 20% de descuento a los clientes cuya compra supere los $1000. Mostrar por pantalla cuál será la cantidad que pagará una persona por su compra.

12) Solicitar la edad de dos hermanos, mostrando un mensaje que informe la edad del mayor y cuántos años de diferencia tiene con el menor.

13) Diseñar una solución que permita ingresar tres números enteros, e informe cuál es el mayor.

14) Dados los lados de un triángulo, informar qué tipo es: isósceles, escaleno o equilátero.

15) Elaborar una solución que permita ingresar un número entero del 1 al 10 y muestre su equivalente en números romanos.

16) Hacer un algoritmo que imprima el nombre de un artículo, clave, precio original y su precio con descuento. El descuento lo hace en base a la clave, si la clave es 01 el descuento es del 10% y si la clave es 02 el descuento en del 20% (solo existen dos claves).

17) Leer 2 números ingresados por el usuario; si son iguales que los multiplique, si el primero es mayor que el segundo que los reste y si no que los sume.

18) Determinar la cantidad de dinero que recibirá un trabajador por concepto de las horas extras trabajadas en una empresa, sabiendo que cuando las horas de trabajo exceden de 40, el resto se consideran horas extras y que estas se pagan al doble de una hora normal cuando no exceden de 8; si las horas extras exceden de 8 se pagan las primeras 8 al doble de lo que se pagan las horas normales y el resto al triple.

19) Una compañía de seguros está abriendo un depto. de finanzas y estableció un programa para captar clientes, que consiste en lo siguiente: Si el monto por el que se efectúa la fianza es menor que $50 000 la cuota a pagar será por el 3% del monto, y si el monto es mayor que $50 000 la cuota a pagar será el 2% del monto. La compañía desea determinar cuál será la cuota que debe pagar un cliente.

20) Se desea escribir un algoritmo que pida la altura de una persona, si la altura es menor o igual a 150 cm envíe el mensaje: “Persona de altura baja”; si la altura está entre 151 y 170 escriba el mensaje: “Persona de altura media” y si la altura es mayor al 171 escriba el mensaje: “Persona alta”.

21) Dado un numero entre 1 y 7 escriba su correspondiente día de la semana así: 1- Lunes 2- Martes 3- Miércoles 4- Jueves 5- Viernes 6- Sábado 7- Domingo

22) Elabore una solución que solicite el ingreso de un número entero distinto de 0 y se informe si es par o no

23) Se desea leer un número de dos cifras, y que se muestre por pantalla sus dígitos. Luego informe si la suma de sus dígitos es múltiplo de 3 o no.

24) En una Granja existen N conejos, N1 blancos y N2 negros. Suponiendo que P1 es el precio de venta de los conejos blancos y P2 es el precio de venta de los conejos negros, imprima el monto total de la venta. Informe al final el color de los conejos que se más se vendieron.

# ESTRUCTURAS REPETITIVAS

25) Un programa que permita al usuario ingresar un número entre el 1 y el 9 y nos muestre la tabla de dicho número hasta su multiplicación por 9. La salida debería ser en caso de que el usuario ingrese el número 2:

| 2x1=2 2x3=6 ....  2x9=18 |
| --- |

26) Obtener un algoritmo que imprima por pantalla todos los números pares del 2 al 300. Utilizar un While.

27) Obtener un algoritmo que imprima por pantalla todos los números comprendidos entre el 5 y el 230, pero en orden inverso. Por ejemplo: 230,229,228,……………….,6,5.

28) Calcular la suma y el promedio de números ingresados por el usuario. La cantidad de números a ingresar no se conoce, pero se sabe que deben pertenecer al rango [0,100]. La carga finaliza cuando el usuario ingresa un valor fuera del rango.

29) Escriba un algoritmo que permita ingresar un Número N positivo y muestre por pantalla la suma de sus dígitos.

30) Dado un número ingresado por el usuario determinar si es primo o no.

31) Escribir un algoritmo que permita ingresar para los N alumnos de una universidad:

a. SEXO ( ‘M’ o ‘F’ ) , edad y carrera ( ‘A’,’B’,’C’ ) .

Imprimir por pantalla la carrera con menor promedio de edad de sus alumnos que son varones.

32) Crea un algoritmo que pida al usuario su código de usuario (un número entero) y su contraseña numérica (otro número entero), y no le permita seguir hasta que introduzca como código 1024 y como contraseña 4567.

33) Elabore un algoritmo que permita mostrar por pantalla los números pares hasta el 18.

34) Se pide representar el algoritmo que nos calcule la suma de los N primeros números impares. Es decir, si ingresamos un 5, nos haga la suma de 1+3+5+7+9.

35) Teniendo en cuenta que la clave es “111MIL”, escribir un algoritmo que nos pida una clave. Solo tenemos 3 intentos para acertar, si fallamos los 3 intentos nos mostrara un mensaje indicándonos que hemos agotado esos 3 intentos. Si acertamos la clave, saldremos del proceso.

36) En una empresa trabajan n empleados cuyos sueldos oscilan entre $100 y $500, realizar un Algoritmo que lea los sueldos que cobra cada empleado e informe cuántos empleados cobran entre $100 y $300 y cuántos cobran más de $300. Además el programa deberá informar el importe que gasta la empresa en sueldos al personal.

37) Crear un programa que muestre todos los números primos entre dos números ingresados por el usuario. Ejemplo si ingresa 1 y 10: 2, 3, 5, 7, 9

# ARRAYS DINÁMICOS

## 38) Inversión de un Array:

Escribe un programa que invierta los elementos de un array dado. Por ejemplo, si el array de entrada es [1, 2, 3, 4, 5], el programa debería retornar [5, 4, 3, 2, 1].

## 39) Concatenación de Arrays:

Crea una función que tome dos arrays como argumentos y retorne un nuevo array que sea la concatenación de los dos arrays de entrada. Por ejemplo, concatenar [1, 2, 3] y [4, 5, 6] debería dar como resultado [1, 2, 3, 4, 5, 6].

## 40) Encontrar el Elemento Máximo:

Desarrolla una función que encuentre el elemento máximo en un array de números. Por ejemplo, para el array [10, 3, 4, 15, 7], la función debe retornar 15.

## 41) Filtrado de Array:

Implementa una función que elimine todos los elementos inferiores a 0.

## 42) Rotación de Arrays:

Escribe un programa que rote los elementos de un array hacia la derecha n veces. Por ejemplo, si el array es [1, 2, 3, 4, 5] y n = 2, el resultado debe ser [4, 5, 1, 2, 3].

## 43) Suma de Subarrays:

Diseña una función que tome un array de números y devuelva un nuevo array donde cada elemento es la suma de los elementos hasta esa posición en el array original. Por ejemplo, para el array [1, 2, 3, 4], el resultado sería [1, 3, 6, 10].

## 44) Eliminación de Duplicados:

Desarrolla un programa que elimine todos los elementos duplicados de un array. Utiliza un método que optimice la búsqueda de duplicados sin usar funciones predefinidas como Set.

## 45) Mezcla de Arrays:

Escribe una función que combine dos arrays de entrada intercalando sus elementos. Por ejemplo, dados los arrays [a, b, c] y [1, 2, 3], el resultado debería ser [a, 1, b, 2, c, 3].

## 46) Array de Pares:

Crea una función que tome un número entero n y retorne un array que contenga pares de números cuya suma sea igual a n. Cada par debe ser único y estar representado como un array. Por ejemplo, si n = 5, algunos pares posibles incluyen [ 2, 3, 4, 1,.... 0, 5 ].

# ARCHIVOS CON LECTURA LÍNEA POR LÍNEA

## 47) Invertir Líneas:

Escribe un programa que lea un archivo de texto línea por línea e invierta el orden de las líneas en el archivo de salida.

## 48) Lista de compras

Escribe un programa que guarde una lista de compras en un archivo línea por línea, es decir que cada ítem de la compra significa una línea nueva en el archivo. El programa debe permitir al usuarios agregar tantos artículos como quiera.

## 49) Contar Vocales

Escribe un programa que lea un archivo de texto carácter por carácter y cuente el número de vocales (a, e, i, o, u) en el archivo.

# MATRICES

## 50) Suma de Elementos de una Matriz

Escribe un programa que calcule la suma de todos los elementos en una matriz de enteros. De una matriz de 3x3

| public static void main(String[] args) {  List<List<Integer>> matriz = new ArrayList<>();  List<Integer> fila1 = new ArrayList<>();  fila1.add(1);  fila1.add(2);  fila1.add(3);  matriz.add(fila1);  List<Integer> fila2 = new ArrayList<>();  fila2.add(4);  fila2.add(5);  fila2.add(6);  matriz.add(fila2);  List<Integer> fila3 = new ArrayList<>();  fila3.add(7);  fila3.add(8);  fila3.add(9);  matriz.add(fila3);  int suma = 0;  for (List<Integer> fila : matriz) {  for (Integer elemento : fila) {  suma += elemento;  }  }  System.out.println("La suma es: " + suma);  } |
| --- |

## 51) Matriz Transpuesta

Escribe un programa que calcule la transpuesta de una matriz dada. La transpuesta de una matriz se obtiene intercambiando sus filas por columnas. De una matriz de 3x3

| int[][] matriz = {  { 1, 2, 3 },  { 4, 5, 6 },  { 7, 8, 9 }  };  int[][] transpuesta = new int[matriz[0].length][matriz.length];  for (int i = 0; i < matriz.length; i++) {  for (int j = 0; j < matriz[i].length; j++) {  transpuesta[j][i] = matriz[i][j];  }  }  System.out.println("Transpuesta:");  for (int i = 0; i < transpuesta.length; i++) {  for (int j = 0; j < transpuesta[i].length; j++) {  System.out.print(transpuesta[i][j] + " ");  }  System.out.println();  } |
| --- |

## 52) Multiplicación de Matrices

Escribe un programa que multiplique dos matrices de enteros. La multiplicación de matrices solo es posible si el número de columnas de la primera matriz es igual al número de filas de la segunda matriz. De una matriz de 3x3.

1 2 3 7 8 **58 54**

4 5 6 x 9 10 =

11 12  **139 154**

| *// Método que multiplica la matriz*  public static int[][] multiplicarMatrices(int[][] m1, int[][] m2) {  int filas1 = m1.length;  int columnas1 = m1[0].length;  int columnas2 = m2[0].length;   int[][] resultado = new int[filas1][columnas2];   for (int i = 0; i < filas1; i++) {  for (int j = 0; j < columnas2; j++) {  for (int k = 0; k < columnas1; k++) {  resultado[i][j] += m1[i][k] \* m2[k][j];  }  }  }  return resultado;  } |
| --- |

## 53) Determinante de una Matriz 2x2

Escribe un programa que calcule el determinante de una matriz 2x2. El determinante de una matriz 2x2 se calcula como: ad - bc, donde la matriz es:

| a b |

| c d |

| public static void main(String[] args) {  int[][] matriz = {  { 4, 3 },  { 2, 1 }  };  int determinante = (matriz[0][0] \* matriz[1][1]) - (matriz[0][1] \* matriz[1][0]);  System.out.println("El determinante 2x2 es: " + determinante);  } |
| --- |

## 54) Diagonal Principal de una Matriz

Escribe un programa que imprima los elementos de la diagonal principal de una matriz cuadrada. De una matriz de 3x3

| public static void main(String[] args) {  int[][] matriz = {  { 1, 2, 3 },  { 4, 5, 6 },  { 7, 8, 9 }  };  System.out.println("Elementos de la diagonal principal:");  for (int i = 0; i < matriz.length; i++) {  System.out.println(matriz[i][i]);  }  } |
| --- |