



# UNIVERSIDAD TECNOLÓGICA DE ESCUINAPA

<b>Carrera:</b> TSU en Tecnologías de la Información, Área Desarrollo de Software Multiplataforma.	<b>Grupo:</b> 4ATI
<b>Asignatura:</b> Estructuras de Datos Aplicadas	<b>Unidad:</b> Arreglos
<b>Profesor:</b> MGTI Jorge Luis De La Paz Ramos	

1. Construya un programa que realice las siguientes acciones. Considere un arreglo ordenado en forma ascendente
  - Genere un arreglo de 100 números aleatorios entre 0 y 100
  - Dado un número ingresado por el usuario, de como resultado la posición de la primera ocurrencia.
  - Dado un número ingresado ver cuantas veces se repite en el arreglo.
  - Muestre el arreglo las veces que el usuario lo requiera.

```
11 public class A2_1 {
12     public static void main(String[] args) {
13         Scanner ent = new Scanner(System.in);
14         int [] num = new int[100];
15         for(int i=0; i<100; i++){
16             num[i] = 1 + (int)(Math.random() * 100);
17         }
18         System.out.println("Ingresa el número que quieras buscar en el arreglo");
19         int n = ent.nextInt();
20         int pos=0, cont =0;
21         for(int i=0; i<num.length; i++){
22             if(n == num[i]){
23                 pos = i;
24                 break;
25             }
26         }
27         if(pos == 0){
28             System.out.println("El número no se encontro en el arreglo");
29         }else{
30             System.out.println("El numero ingresado se encontro por primera vez en la posición ["+pos+"]");
31         }
32         for(int i=0; i<num.length; i++){
33             if(n == num[i]){
34                 cont++;
35             }
36         }
37         if(cont > 0){
38             System.out.println("Y se repite "+cont+" veces en el arreglo");
39         }
40         System.out.println("¿Deseas ver todos los números en el arreglo? s/n");
41         String r = ent.nextLine();
42         if(r.equals("s")){
43             String r2=ent.nextLine();
44             do{
45                 for(int i=0; i<num.length; i++){
46                     System.out.println("num["+i+"] = "+num[i]);
47                 }
48                 System.out.println("¿Quiere volver a imprimir? s/n");
49                 r2 = ent.nextLine();
50             }while(r2.equals("s"));
51         }else{
52             System.out.println("oki");
53         }
54     }
55 }
```



## UNIVERSIDAD TECNOLÓGICA DE ESCUINAPA

Ingresa el número que quieras buscar en el arreglo  
44

El número no se encontro en el arreglo

¿Deseas ver todos los números en el arreglo? s/n

s

```
num[0] = 64
num[1] = 56
num[2] = 91
num[3] = 68
num[4] = 23
num[5] = 23
num[6] = 65
num[7] = 33
num[8] = 38
num[9] = 52
num[10] = 36
num[11] = 49
num[12] = 96
num[13] = 19
num[14] = 7
num[15] = 60
num[16] = 1
num[17] = 14
num[18] = 86
num[19] = 78
num[20] = 5
num[21] = 76
num[22] = 59
num[23] = 71
num[24] = 50
num[25] = 79
num[26] = 7
num[27] = 60
num[28] = 1
num[29] = 36
num[30] = 91
num[31] = 73
num[32] = 76
num[33] = 24
num[34] = 37
num[35] = 14
num[36] = 30
```



# UNIVERSIDAD TECNOLÓGICA DE ESCUINAPA

2. Una compañía distribuye N productos a distintos comercios de la ciudad. Para ello almacena en un arreglo toda la información relacionada con su mercancía:
- Clave
  - Descripción
  - Existencia
  - Mínimo para mantener de existencia.
  - Precio Unitario.

```
8  public class A2 {  
9      public static void main(String[] args) {  
10         Scanner ent = new Scanner(System.in);  
11  
12         System.out.print("Cantidad de Productos: ");  
13         int n = ent.nextInt();  
14         ent.nextLine();  
15         String[] clave = new String[n];  
16         String[] descripcion = new String[n];  
17         int[] existencia = new int[n];  
18         int[] minimo = new int[n];  
19         double[] precio = new double[n];  
20         for (int i = 0; i < n; i++) {  
21             System.out.println("|-- Producto " + (i + 1) + " --|");  
22             System.out.print("Clave: ");  
23             clave[i] = ent.nextLine();  
24             System.out.print("Descripción: ");  
25             descripcion[i] = ent.nextLine();  
26             System.out.print("Productos en Existencia: ");  
27             existencia[i] = ent.nextInt();  
28             System.out.print("Stock Mínimo para mantener Existencia: ");  
29             minimo[i] = ent.nextInt();  
30             System.out.print("Precio Unitario: ");  
31             precio[i] = ent.nextDouble();  
32             ent.nextLine();  
33         }  
34         System.out.println("<-- Productos Registrados -->");  
35         for (int i = 0; i < n; i++) {  
36             System.out.println("Producto " + (i + 1) + ":-");  
37             System.out.println("Clave: " + clave[i]);  
38             System.out.println("Descripción: " + descripcion[i]);  
39             System.out.println("En Existencia: " + existencia[i]);  
40             System.out.println("Min. para mantener Existencia: " + minimo[i]);  
41             System.out.println("Precio Unitario: " + precio[i]);  
42         }  
43     }  
44 }  
45  
run:  
Cantidad de Productos: 2  
|-- Producto 1 --|  
Clave: 123  
Descripción: nose hola  
Productos en Existencia: 500  
Stock Mínimo para mantener Existencia: 1  
Precio Unitario: 100  
|-- Producto 2 --|  
Clave: 321  
Descripción: aloh eson  
Productos en Existencia: 005  
Stock Mínimo para mantener Existencia: 1  
Precio Unitario: 001  
<-- Productos Registrados -->  
Producto 1:-  
Clave: 123  
Descripción: nose hola  
En Existencia: 500  
Min. para mantener Existencia: 1  
Precio Unitario: 100.0  
Producto 2:-  
Clave: 321  
Descripción: aloh eson  
En Existencia: 5  
Min. para mantener Existencia: 1  
Precio Unitario: 1.0  
BUILD SUCCESSFUL (total time: 1 minute 7 seconds)
```



# UNIVERSIDAD TECNOLÓGICA DE ESCUINAPA

3. Escriba un programa que efectúe las siguientes operaciones:
- Venta de un producto: se deben actualizar los campos que correspondan y verificar que la nueva existencia no esté por debajo del mínimo. (Datos: clave, cantidad vendida)
  - Reabastecimiento de un producto: se deben actualizar los campos que correspondan. (Datos: clave, cantidad comprada.)
  - Actualizar el precio de un producto. (Datos: clave, porcentaje de aumento.)
  - Informar sobre un productor: se deben proporcionar todos los datos relacionados con un producto. (Dato: clave)

```
5  public class A3 {
6      public static void main(String[] args) {
7          Scanner ent = new Scanner(System.in);
8
9          System.out.print("Cuantos productos vas a tener? ");
10         int n = ent.nextInt();
11
12         int[] c = new int[n];
13         String[] no = new String[n];
14         int[] ex = new int[n];
15         int[] mi = new int[n];
16         double[] pre = new double[n];
17
18         // Llenar los productos
19         System.out.println("Ahora llena los datos de cada producto:");
20         for(int i = 0; i < c.length; i++) {
21             System.out.println("Producto " + (i + 1) + ":");
22             System.out.print("Clave: ");
23             c[i] = ent.nextInt();
24             ent.nextLine();
25             System.out.print("Nombre: ");
26             no[i] = ent.nextLine();
27             System.out.print("Existencias: ");
28             ex[i] = ent.nextInt();
29             System.out.print("Mínimo: ");
30             mi[i] = ent.nextInt();
31             System.out.print("Precio: ");
32             pre[i] = ent.nextDouble();
33         }
34
35         System.out.println("Tienda");
36
37
38         while(true) {
39             System.out.println("1. Vender");
40             System.out.println("2. Reabastecimiento ");
41             System.out.println("3. Cambiar precio");
42             System.out.println("4. Ver producto");
43             System.out.println("5. Salir");
44             System.out.print("Elige: ");
45             int opcion = ent.nextInt();
46
47             if(opcion == 5) break;
48
49             System.out.print("Clave del producto: ");
```



# UNIVERSIDAD TECNOLÓGICA DE ESCUINAPA

```
49     System.out.print("Clave del producto: ");
50     int claveB = ent.nextInt();
51
52     // Buscar el producto
53     int pro = -1;
54     for(int i = 0; i < c.length; i++) {
55         if(c[i] == claveB) {
56             pro = i;
57             break;
58         }
59     }
60
61     if(pro == -1) {
62         System.out.println("No existe ese producto");
63         continue;
64     }
65
66     if(opcion == 1) {
67         System.out.print("Cuántos vender: ");
68         int cant = ent.nextInt();
69         if(ex[pro] >= cant) {
70             ex[pro] -= cant;
71             System.out.println("Vendido. Quedan: " + ex[pro]);
72         } else {
73             System.out.println("No hay suficientes");
74         }
75     } else if(opcion == 2) {
76         System.out.print("Cuántos comprar: ");
77         int cant = ent.nextInt();
78         ex[pro] += cant;
79         System.out.println("Comprado. Ahora hay: " + ex[pro]);
80     } else if(opcion == 3) {
81         System.out.print("Nuevo precio: $");
82         pre[pro] = ent.nextDouble();
83         System.out.println("Precio actualizado");
84     } else if(opcion == 4){
85         System.out.println("Producto: " + no[pro]);
86         System.out.println("Existencias: " + ex[pro]);
87         System.out.println("Precio: $" + pre[pro]);
88     }
89
90
91 }
92
93 }
```



# UNIVERSIDAD TECNOLÓGICA DE ESCUINAPA

```
run:  
Cuantos productos vas a tener? 1  
Ahora llena los datos de cada producto:  
Producto 1:  
Clave: 100  
Nombre: pizzass  
Existencias: 2  
Minimo: 1  
Precio: 100  
Tienda  
1. Vender  
2. Reabastecimiento  
3. Cambiar precio  
4. Ver producto  
5. Salir  
Elige: 3  
Clave del producto: 100  
Nuevo precio: $200  
Precio actualizado  
1. Vender  
2. Reabastecimiento  
3. Cambiar precio  
4. Ver producto  
5. Salir  
Elige: 4  
Clave del producto: 100  
Producto: pizzass  
Existencias: 2  
Precio: $200.0  
1. Vender  
2. Reabastecimiento  
3. Cambiar precio  
4. Ver producto  
5. Salir
```

4. Al momento de su ingreso al hospital, a un paciente se le solicitan los siguientes datos:

- Nombre
- Edad
- Sexo
- Domicilio:
  - Calle
  - Número
  - Ciudad
- Teléfono
- Seguro (este campo tendrá el valor VERDADERO si el paciente tiene seguro médico y FALSO en otro caso).

Escriba un programa que pueda llevar a cabo las siguientes operaciones:

- a) Listar los nombres de todos los pacientes hospitalizados.
- b) Obtener el porcentaje de pacientes hospitalizados en las siguientes categorías (dadas por la edad):
  - a. Niños: hasta 13 años.
  - b. Jóvenes: mayores de 13 años y menores de 30
  - c. Adultos: mayores de 30 años.
- c) Obtener el porcentaje de hombres y de mujeres hospitalizados.
- d) Dado el nombre de un paciente, listar todos los datos relacionados con dicho paciente.



# UNIVERSIDAD TECNOLÓGICA DE ESCUINAPA

```
3  public class A4 {
4      public static void main(String[] args) {
5          Scanner ent = new Scanner(System.in);
6          String[] nombre= new String[1];
7          int[] edad=new int[1];
8          String[] sexo=new String[1];
9          String[] calle=new String[1];
10         String[] num=new String[1];
11         String[] ciudad=new String[1];
12         String[] tel=new String[1];
13         boolean[] seguro=new boolean[1];
14         System.out.println("Nombre: ");
15         nombre[0]=ent.nextLine();
16         System.out.println("Edad: ");
17         edad[0]=ent.nextInt();
18         System.out.println("Sexo: ");
19         sexo[0]=ent.nextLine();
20         System.out.println("Ciudad: ");
21         ciudad[0]=ent.nextLine();
22         System.out.println("Calle: ");
23         calle[0]=ent.nextLine();
24         System.out.println("Número: ");
25         num[0]=ent.nextLine();
26         System.out.println("Telefono: ");
27         tel[0]=ent.nextLine();
28         System.out.println("Seguro: ".toUpperCase());
29         String r= ent.nextLine();
30         if(r.equalsIgnoreCase("SI") || r.equalsIgnoreCase("Sí")){
31             seguro[0]= true;
32         }else{
33             seguro[0]= false;
34         }
35         System.out.println("-Datos-");
36         System.out.println("Nombre: "+nombre[0]);
37         System.out.println("Edad: "+edad[0]);
38         System.out.println("Sexo: "+sexo[0]);
39         System.out.println("Domicilio: ");
40         System.out.println("Calle: "+calle[0]);
41         System.out.println("Número: "+num[0]);
42         System.out.println("Telefono: "+tel[0]);
43         System.out.println("Seguro: "+seguro[0]);
44     }
45 }
```



# UNIVERSIDAD TECNOLÓGICA DE ESCUINAPA

```
50 //Opciones
51 String res="Sí";
52 while(res.equalsIgnoreCase("Sí") || res.equalsIgnoreCase("SI")){
53     System.out.println("\nQue desea hacer?\n(a) Listar.\nb) Porcentaje por Edad."
54             + "\nc) Porcentaje por sexo.\nd) Mostrar datos de un paciente.".toUpperCase());
55     String res2 =ent.nextLine();
56     if(res2.equalsIgnoreCase("A")){
57         System.out.println("\nLista de Pacientes: \n");
58         for(int i=0; i<nombre.length;i++){
59             System.out.println("<-Paciente "+(i+1)+"->");
60             System.out.println("Nombre: "+nombre[i]+"\n");
61         }
62     }else if(res2.equalsIgnoreCase("B")){
63         int nN=0;
64         int nJ=0;
65         int nA=0;
66         System.out.println("\nPorcentajes de pacientes por grupos de edad: \n");
67         for(int i=0; i<nombre.length; i++){
68             if(edad[i]<13){
69                 nN++;
70             }else if(edad[i]<30){
71                 nJ++;
72             }else{
73                 nA++;
74             }
75         }
76         double pNiños=(100.0*nN)/nombre.length;
77         double pJovenes=(100.0*nJ)/nombre.length;
78         double pAdultos=(100.0*nA)/nombre.length;
79         System.out.println("Niños(0 a 13 años): "+pNiños+"%."
80             + "\nJovenes(14 a 29): "+pJovenes+"%."
81             + "\nAdultos(30 años en adelante): "+pAdultos+"%.");
82     }
83     else if(res2.equalsIgnoreCase("C")){
84         int nF=0;
85         int nM=0;
86         System.out.println("\nPorcentajes de pacientes por grupos de Genero: \n");
87         for(int i=0; i<nombre.length; i++){
88             if(sexo[i].equalsIgnoreCase("F")){
89                 nF++;
90             }else{
91                 nM++;
92             }
93         }
94         double pF=(100.0*nF)/nombre.length;
95         double pM=(100.0*nM)/nombre.length;
96         System.out.println("Mujeres: "+pF+"%."
97             + "\nHombres: "+pM+"%.");
98     }else if(res2.equalsIgnoreCase("D")){
99         System.out.println("\n<-Buscar un paciente->\n");
100        System.out.println("Nombre del paciente: ");
101        String nom=ent.nextLine();
102        int j;
103        for(j=0; j<nombre.length; j++){
104            if(nombre[j].equalsIgnoreCase(nom)){
105                if(nombre[j].equalsIgnoreCase(nom)){
106                    System.out.println("-Datos-");
107                    System.out.println("Nombre: "+nombre[j]);
108                    System.out.println("Edad: "+edad[j]);
109                    System.out.println("Sexo: "+sexo[j]);
110                    System.out.println("Domicilio.");
111                    System.out.println("Ciudad: "+ciudad[j]);
112                    System.out.println("Calle: "+cal[j]);
113                    System.out.println("Número: "+num[j]);
114                    System.out.println("Teléfono: "+tel[j]);
115                    System.out.println("Seguro: "+seguro[j]);
116                    break;
117                }
118            }
119            if(j==nombre.length){
120                System.out.println("Paciente"+nom+" no encontrado.");
121            }
122        }
123    }else{
124        System.out.println("Opción no validad.");
125    }
126    System.out.println("\nDesea realizar algo más?(Sí/NO)");
127    res=ent.nextLine();
128 }
129 }
130 }
```



# UNIVERSIDAD TECNOLÓGICA DE ESCUINAPA

Cantidad de Pacientes:

1

Nombre:

Sofia

Edad:

20

Sexo:

F

Domicilio.

Ciudad:

Rosario

Calle:

B

Número:

00

Telefono:

00000000

SEGURO:

SI

¿Que desea hacer?

a) Listar.

b) Porcentaje por Edad.

C) PORCENTAJE POR SEXO.

D) MOSTRAR DATOS DE UN PACIENTE.

A

Lista de Pacientes:

<-Paciente 1->

Nombre: Sofia

¿Desea realizar algo más?(SÍ/NO)

BUILD SUCCESSFUL (total time: 25 seconds)

I

5. Supongamos que en una elección hubo 12 candidatos (con identificadores 1,2,3, ..., 12). Por otra parte, los votos para cada candidato se teclean de manera desorganizada como se muestra a continuación:

1 5 7 5 1 12 10 7 1 7 5 8 1 5 -1(número negativo para detener la captura de votos)

Construya un programa que pueda proporcionar la siguiente información:

- El número de votos de cada candidato al final de la elección.
- El candidato ganador, el número de votos que obtuvo y el porcentaje correspondiente del total de la elección. Suponemos que el candidato ganador no empató en número de votos con otro candidato.



# UNIVERSIDAD TECNOLÓGICA DE ESCUINAPA

```
13  public static void main(String[] args) {
14      Scanner ent = new Scanner(System.in);
15      int[] votos = new int[12]; // candidatos y votos respectivos
16      int totV = 0;
17      System.out.println("Captura de votos (numero(-) para terminar):");
18      while (true) {
19          System.out.print("Voto para candidato (1-12): ");
20          int voto = ent.nextInt();
21          ent.nextLine();
22
23          if (voto < 0) {
24              break; // detener si es <0
25          } else if (voto >= 1 && voto <= 12) {
26              votos[voto - 1]++; // aumentar la cantidad de votos en esa posición(-1 para ajustar)
27              totV++; // Votos en total ++
28          } else {
29              System.out.println("Número de candidato no válido. Ingrese entre 1 y 12.");
30          }
31
32          // Votos por candidato
33          System.out.println("\nResultados:");
34          for (int i = 0; i < votos.length; i++) {
35              System.out.println("Candidato " + (i + 1) + ": " + votos[i] + " votos");
36          }
37
38          // Ganador
39          int maxVotos = 0;
40          int ganador = 0;
41          for (int i = 0; i < votos.length; i++) {
42              if (votos[i] > maxVotos) {
43                  maxVotos = votos[i];
44                  ganador = i;
45              }
46
47          // Calcular el porcentaje
48          double porc = (100.0 * maxVotos) / totV;
49          // Mostrar ganador
50          System.out.println("\nCandidato ganador: " + (ganador+1));
51          System.out.println("Votos obtenidos: " + maxVotos);
52          System.out.println("Porcentaje del total: " + porc + "%");
53      }
54  }
```

run:

```
Captura de votos (numero(-) para terminar):
Voto para candidato (1-12): 5
Voto para candidato (1-12): 6
Voto para candidato (1-12): 8
Voto para candidato (1-12): 7
Voto para candidato (1-12): -8
```

Resultados:

```
Candidato 1: 0 votos
Candidato 2: 0 votos
Candidato 3: 0 votos
Candidato 4: 0 votos
Candidato 5: 1 votos
Candidato 6: 1 votos
Candidato 7: 7 votos
Candidato 8: 1 votos
Candidato 9: 0 votos
Candidato 10: 0 votos
Candidato 11: 0 votos
Candidato 12: 0 votos
```

```
Candidato ganador: 7
```

```
Votos obtenidos: 7
```

```
Porcentaje del total: 70.0%
```

```
BUILD SUCCESSFUL (total time: 15 seconds)
```

6. Almacene en un arreglo las temperaturas de un día (0-23 horas). Calcule su media e imprimir la temperatura más alta y la más baja; asimismo imprimir la hora respectiva. Por ejemplo, la temperatura media fue de 21.5 grados Celsius, la más alta de 29 grados Celsius a las 14 horas y la más baja de 9.4 grados Celsius a las 3 horas.



# UNIVERSIDAD TECNOLÓGICA DE ESCUINAPA

```
12
13     public static void main(String[] args) {
14         Scanner ent = new Scanner(System.in);
15         double[] temperaturas= new double[24];
16         for(int i=0; i<temperaturas.length; i++){
17             System.out.println((i+1)+" hrs, Temperatura: ");
18             temperaturas[i]=ent.nextDouble();
19         }
20         double maxTemp = temperaturas[0];
21         int hrM = 0;
22         double minTemp=temperaturas[0];
23         int hrm=0;
24         double media=0;
25         for (int i = 0; i < temperaturas.length; i++) {
26             if (temperaturas[i] > maxTemp) {
27                 maxTemp = temperaturas[i];
28                 hrM = i;
29             }
30             if(temperaturas[i] < minTemp){
31                 minTemp = temperaturas[i];
32                 hrm = i;
33             }
34             media=media+temperaturas[i];
35         }
36         media=media/temperaturas.length;
37         System.out.println("Media: "+media+"°C.");
38         System.out.println("Temperatura más alta: "+maxTemp+"°C, a las "+(hrM+1)+" hrs");
39         System.out.println("Temperatura más baja: "+minTemp+"°C, a las "+(hrm+1)+" hrs");
40     }
41 }
```

18 hrs, Temperatura:  
43  
19 hrs, Temperatura:  
43  
20 hrs, Temperatura:  
43  
21 hrs, Temperatura:  
43  
22 hrs, Temperatura:  
43  
23 hrs, Temperatura:  
43  
24 hrs, Temperatura:  
34  
Media: 43.625°C.  
Temperatura más alta: 56.0°C, a las 1 hrs  
Temperatura más baja: 34.0°C, a las 24 hrs  
**BUILD SUCCESSFUL (total time: 1 minute 11 seconds)**

7. Almacene en un arreglo la temperatura de cada día de una determinada semana y que realice lo siguiente:
- La temperatura promedio.
  - Un arreglo que contenga las diferencias de cada temperatura con respecto al promedio.
  - La menor temperatura y el número de día en que ocurrió.
  - La mayor temperatura y el número de día en que ocurrió.



# UNIVERSIDAD TECNOLÓGICA DE ESCUINAPA

```
12 public static void main(String[] args) {
13
14     Scanner ent = new Scanner(System.in);
15     double[] temperaturas= new double[7];
16     double[] dif=new double[temperaturas.length];
17     for(int i=0; i<temperaturas.length; i++){
18         System.out.println((i+1)+" hrs, Temperatura: ");
19         temperaturas[i]=ent.nextDouble();
20     }
21     double maxTemp = temperaturas[0];
22     int dM = 0;
23     double minTemp=temperaturas[0];
24     int dm=0;
25     double media=0;
26     for (int i = 0; i < temperaturas.length; i++) {
27         if (temperaturas[i] > maxTemp) {
28             maxTemp = temperaturas[i];
29             dM = i;
30         }
31         if(temperaturas[i] < minTemp){
32             minTemp = temperaturas[i];
33             dm = i;
34         }
35         media=media+temperaturas[i];
36     }
37     media=media/temperaturas.length;
38     System.out.println("Tem. Promedio: "+media+"°C.");
39     for(int i=0; i<dif.length; i++){
40         dif[i]=temperaturas[i]-media;
41         System.out.println("Día "+(i+1)+", dif. entre temp. correspondiente y temp. promedio: "+dif[i]);
42     }
43     System.out.println("Temperatura más alta: "+maxTemp+"°C, Dia: "+(dM+1));
44     System.out.println("Temperatura más baja: "+minTemp+"°C, Dia: "+(dm+1));
45 }
}
/nrs, temperatura:
36
Tem. Promedio: 33.285714285714285°C.
Día 1, dif. entre temp. correspondiente y temp. promedio: 0.7142857142857153
Día 2, dif. entre temp. correspondiente y temp. promedio: 0.7142857142857153
Día 3, dif. entre temp. correspondiente y temp. promedio: -10.285714285714285
Día 4, dif. entre temp. correspondiente y temp. promedio: 0.7142857142857153
Día 5, dif. entre temp. correspondiente y temp. promedio: 2.7142857142857153
Día 6, dif. entre temp. correspondiente y temp. promedio: 2.7142857142857153
Día 7, dif. entre temp. correspondiente y temp. promedio: 2.7142857142857153
Temperatura más alta: 36.0°C, Dia: 5
Temperatura más baja: 23.0°C, Dia: 3
```

## 8. Almacene en un arreglo, 20 elementos de tipo entero e imprimir:

- La suma de los elementos que ocupan posiciones pares.
- El mayor de los elementos que ocupan posiciones impares.
- La posición del mayor número par.

```
12 public static void main(String[] args) {
13
14     Scanner ent = new Scanner(System.in);
15     int[] num= new int[20];
16     int sum=0;
17     int numMi=num[0];
18     int numMp=num[0];
19     for(int i=0; i<num.length; i++){
20         System.out.println("Ingresa un número entero: ");
21         num[i]=ent.nextInt();
22         if(num[i]%2==0){
23             sum=sum+num[i];
24             if(num[i]>numMp){
25                 numMp=num[i];
26             }
27         }else{
28             if(num[i]>numMi){
29                 numMi=num[i];
30             }
31         }
32     }
33     System.out.println("Suma de elementos pares: "+sum);
34     System.out.println("Mayor número impar: "+numMi);
35     System.out.println("Mayor número par: "+numMp);
36 }
}

```



## UNIVERSIDAD TECNOLÓGICA DE ESCUINAPA

```
Ingresá un número entero:  
8  
Suma de elementos pares: 160  
Mayor número impar: 0  
Mayor número par: 8  
BUILD SUCCESSFUL (total time: 17 seconds)
```



## UNIVERSIDAD TECNOLÓGICA DE ESCUINAPA

9. Almacene en un arreglo, n elementos (máximo 30) e imprimir la suma:
- de números pares.
  - de números impares.
  - total de los elementos del arreglo.

The screenshot shows a Java code editor with the following code:

```
13 public static void main(String[] args) {  
14     Scanner ent = new Scanner(System.in);  
15     int n=0;  
16     while(n<=0 || n>30){  
17         System.out.println("Cantidad de elementos: ");  
18         n= ent.nextInt();  
19     }  
20     int[] elem= new int[n];  
21     int sumI=0;  
22     int sumP=0;  
23     for(int i=0; i<elem.length; i++){  
24         System.out.println((i+1)+" elemento: ");  
25         elem[i]= ent.nextInt();  
26         if(elem[i]%2==0){  
27             sumP=sumP+elem[i];  
28         }else{  
29             sumI=sumI+elem[i];  
30         }  
31     }  
32     System.out.println("Suma elementos pares: "+sumP);  
33     System.out.println("Suma elementos impares: "+sumI);  
34     System.out.println("Cantidad total de elementos: "+elem.length);  
35 }  
36 }
```

Below the code, the terminal output shows the execution of the program:

```
Cantidad de elementos:  
4  
1 elemento:  
4  
2 elemento:  
6  
3 elemento:  
5  
4 elemento:  
8  
Suma elementos pares: 18  
Suma elementos impares: 5  
Cantidad total de elementos: 4  
BUILD SUCCESSFUL (total time: 9 seconds)
```

10. Almacene en un arreglo 15 números e imprimir cuántos son ceros, cuántos son negativos y cuántos positivos. Imprimir además la suma de los negativos y la suma de los positivos.



# UNIVERSIDAD TECNOLÓGICA DE ESCUINAPA

```
12 public static void main(String[] args) {  
13     Scanner ent = new Scanner(System.in);  
14     int[] num= new int[15];  
15     int cN=0, sumN=0;  
16     int cP=0, sumP=0;  
17     int cC=0;  
18     for(int i=0; i<num.length; i++){  
19         System.out.println("Ingrese un número: ");  
20         num[i]=ent.nextInt();  
21         if(num[i]>0){  
22             cP++;  
23             sumP=sumP+num[i];  
24         }else if(num[i]<0){  
25             cN++;  
26             sumN=sumN+num[i];  
27         }else{  
28             cC++;  
29         }  
30     }  
31     System.out.println("Cantidad de 0: "+cC);  
32     System.out.println("Cantidad de num(-): "+cN+". Suma: "+sumN);  
33     System.out.println("Cantidad de num(+): "+cP+". Suma: "+sumP);  
34 }  
35 }  
  
Ingrese un número:  
9  
Ingrese un número:  
8  
Cantidad de 0: 0  
Cantidad de num(-): 0. Suma: 0  
Cantidad de num(+): 15. Suma: 126  
RUTINA SUCESOS (total time: 18 seconds)
```

11. Inicialice un arreglo con 10 valores. Genere un arreglo a con los números negativos y otro arreglo b con los positivos o iguales a cero.

```
13 public static void main(String[] args) {  
14     Scanner ent = new Scanner(System.in);  
15     int[] numeros = new int[10];  
16     System.out.println("Ingrese 10 números (positivos, negativos o cero):");  
17     for (int i = 0; i < numeros.length; i++) {  
18         System.out.print("Número " + (i + 1) + ": ");  
19         numeros[i] = ent.nextInt();  
20     }  
21     int contNeg = 0;  
22     int contPos = 0;  
23     for (int i = 0; i < numeros.length; i++) {  
24         if (numeros[i] < 0) {  
25             contNeg++;  
26         } else {  
27             contPos++;  
28         }  
29     }  
30     int[] a = new int[contNeg];  
31     int[] b = new int[contPos];  
32     int j = 0; // índice para -  
33     int k = 0; // índice para +  
34     for (int i = 0; i < numeros.length; i++) {  
35         if (numeros[i] < 0) {  
36             a[j] = numeros[i];  
37             j++;  
38         } else {  
39             b[k] = numeros[i];  
40             k++;  
41         }  
42     }  
43     System.out.println("\nArreglo original:");  
44     for (int i = 0; i < numeros.length; i++) {  
45         System.out.print(numeros[i] + " ");  
46     }  
47     System.out.println("\n\nArreglo A (números negativos):");  
48     for (int i = 0; i < a.length; i++) {  
49         System.out.print(a[i] + " ");  
50     }  
51     System.out.println("\n\nArreglo B (números positivos o iguales a cero):");  
52     for (int i = 0; i < b.length; i++) {  
53         System.out.print(b[i] + " ");  
54     }  
55 }  
56 }  
57 }
```



# UNIVERSIDAD TECNOLÓGICA DE ESCUINAPA

```
run:
Ingrese 10 números (positivos, negativos o cero):
Número 1: -9
Número 2: -8
Número 3: -7
Número 4: 9
Número 5: 8
Número 6: 8
Número 7: 9
Número 8: 0
Número 9: 90
Número 10: -3

Arreglo original:
-9 -8 -7 9 8 8 9 0 90 -3

Arreglo A (números negativos):
-9 -8 -7 -3

Arreglo B (números positivos o iguales a cero):
9 8 8 9 0 90
BUILD SUCCESSFUL (total time: 26 seconds)
```

12. Almacene en un arreglo n elementos de tipo real (máximo 25) e imprima los siguientes valores:

- Máximo.
- Mínimo.
- La media de los que estén en posiciones pares.
- La varianza. (<http://www.disfrutalasmatematicas.com/datos/desviacion-estandar.html>)

```
4 public static void main(String[] args) {
5     Scanner ent = new Scanner(System.in);
6     System.out.println("Cantidad de Elementos (maximo 25): ");
7     int n = ent.nextInt();
8     if (n > 25) {
9         n = 25;
10        System.out.println("Dije Maximo 25(así se queda).");
11    }
12    double[] numeros = new double[n];
13    for (int i = 0; i < numeros.length; i++) {
14        System.out.print("Número real " + (i + 1) + ": ");
15        numeros[i] = ent.nextDouble();
16    }
17    double max = numeros[0];
18    double min = numeros[0];
19    double sumaPares = 0;
20    int contPares = 0;
21    double sumaTotal = 0;
22    for (int i = 0; i < numeros.length; i++) {
23        if (numeros[i] > max) {
24            max = numeros[i];
25        }
26        if (numeros[i] < min) {
27            min = numeros[i];
28        }
29        if (i % 2 == 0) {
30            sumaPares += numeros[i];
31            contPares++;
32        }
33        sumaTotal += numeros[i];
34    }
35    double mediaPares = sumaPares / contPares;
36    double mediaTotal = sumaTotal / numeros.length;
37    //Varianza
38    double sumaVarianza = 0;
39    for (int i = 0; i < numeros.length; i++) {
40        sumaVarianza += Math.pow(numeros[i] - mediaTotal, 2);
41    }
42    double varianza = sumaVarianza / numeros.length;
43    System.out.println("\nNúmero maximo: " + max);
44    System.out.println("Número minimo: " + min);
45    System.out.println("Media de los valores en posiciones pares: " + mediaPares);
46    System.out.println("Varianza de los valores: " + varianza);
47}
48}
49}
```



# UNIVERSIDAD TECNOLÓGICA DE ESCUINAPA

```
Output [ARRREGLO] Run [Run] ^

run:
Cantidad de Elementos (maximo 25):
4
Número real 1: 20
Número real 2: 21
Número real 3: 22
Número real 4: 23

Número maximo: 23.0
Número minimo: 20.0
Media de los valores en posiciones pares: 21.0
Varianza de los valores: 1.25
BUILD SUCCESSFUL (total time: 12 seconds)
```

13. Almacene en un arreglo 10 números y cambie algún número por otro del mismo arreglo, realice el cambio y muestre el arreglo modificado.

```
12  public static void main(String[] args) {
13      Scanner ent = new Scanner(System.in);
14      int[] num = new int[10];
15
16      for (int i = 0; i < num.length; i++) {
17          System.out.println("Ingresa el " + (i + 1) + ".º número: ");
18          num[i] = ent.nextInt();
19      }
20      System.out.println("\nArreglo Original: " + Arrays.toString(num) + "\n");
21
22      System.out.print("Elige una posición del 0 al 9: ");
23      int n = ent.nextInt();
24      System.out.print("Elige una segunda posición del 0 al 9: ");
25      int n2 = ent.nextInt();
26
27      if (n < 0 || n >= num.length || n2 < 0 || n2 >= num.length) {
28          n=(int) (Math.random() * 10) + 1;
29          n2=(int) (Math.random() * 10) + 1;
30          System.out.println("Posición no válida. Serán "+n+" y "+n2);
31      }
32      int temp = num[n];
33      num[n] = num[n2];
34      num[n2] = temp;
35      System.out.println("\nArreglo modificado: " + Arrays.toString(num));
36  }
```



# UNIVERSIDAD TECNOLÓGICA DE ESCUINAPA



```
run:  
Ingresa el 1º número:  
10  
Ingresa el 2º número:  
10  
Ingresa el 3º número:  
1  
Ingresa el 4º número:  
10  
Ingresa el 5º número:  
2  
Ingresa el 6º número:  
10  
Ingresa el 7º número:  
12  
Ingresa el 8º número:  
11  
Ingresa el 9º número:  
111  
Ingresa el 10º número:  
1333  
  
Arreglo Original: [10, 10, 1, 10, 2, 10, 12, 11, 111, 1333]  
  
Elige una posición del 0 al 9: 8  
Elige una segunda posición del 0 al 9: 2  
  
Arreglo modificado: [10, 10, 111, 10, 2, 10, 12, 11, 1, 1333]  
BUILD SUCCESSFUL (total time: 22 seconds)
```

14. Genere aleatoriamente un arreglo de tamaño 20 con números comprendidos entre 1 y 20 e imprimir:
- Suma de los elementos.
  - Promedio de los elementos.
  - Varianza y desviación estándar de los valores.
  - Elemento que más se repite (moda).



# UNIVERSIDAD TECNOLÓGICA DE ESCUINAPA

```
12 public static void main(String[] args) {
13     Random rand = new Random();
14     int[] num = new int[20];
15     int suma = 0;
16     for (int i = 0; i < num.length; i++) {
17         num[i] = rand.nextInt(20) + 1; // 1 a 20
18         suma += num[i];
19     }
20     System.out.println("Arreglo: " + Arrays.toString(num));
21     //Suma
22     System.out.println("\nSuma de los elementos: " + suma);
23     //Promedio
24     double promedio = (double) suma / num.length;
25     System.out.println("Promedio: " + promedio);
26     //Varianza y desviación estandar
27     double sumaVarianza = 0;
28     for (int i = 0; i < num.length; i++) {
29         sumaVarianza = sumaVarianza + Math.pow(num[i] - promedio, 2);
30     }
31     double varianza = sumaVarianza / num.length;
32     double desviacion = Math.sqrt(varianza);
33     System.out.println("Varianza: " + varianza);
34     System.out.println("Desviación estandar: " + desviacion);
35     int moda = num[0];
36     int maxRepet = 0;
37     for (int i = 0; i < num.length; i++) {
38         int repet = 0;
39         for (int j = 0; j < num.length; j++) {
40             if (num[i] == num[j]) {
41                 repet++;
42             }
43         }
44         if (repet > maxRepet) {
45             maxRepet = repet;
46             moda = num[i];
47         }
48     }
49     System.out.println("Moda: " + moda +
50                         " (repetido " + maxRepet + " veces)");
51 }
52 }
```

Output - ARREGLOS2 (run) ×

run:  
Arreglo: [4, 8, 18, 14, 13, 16, 18, 4, 12, 2, 2, 7, 2, 4, 3, 20, 2, 8, 19, 3]  
Suma de los elementos: 179  
Promedio: 8.95  
Varianza: 41.547500000000014  
Desviación estandar: 6.445735024029456  
Moda: 2 (repetido 4 veces)  
BUILD SUCCESSFUL (total time: 0 seconds)



## UNIVERSIDAD TECNOLÓGICA DE ESCUINAPA

15. Almacene en un arreglo de tamaño 10 números reales. Calcule el promedio e indique cuántos elementos del arreglo son mayores que el promedio y genere otro arreglo con los menores o iguales.

```
11  public class A15 {
12  ┌─ public static void main(String[] args) {
13  ┌─ Scanner ent = new Scanner(System.in);
14  ┌─ int [] num = new int[10];
15  ┌─ int suma=0;
16  ┌─
17  ┌─ for(int i=0; i<num.length; i++){
18  ┌─     System.out.println("Ingresa el valor en la posición ["+i+"]");
19  ┌─     num[i] = ent.nextInt();
20  ┌─     suma = suma+num[i];
21  ┌─ }
22  ┌─ int prom = suma/num.length;
23  ┌─ System.out.println("Promedio = "+prom);
24  ┌─ System.out.println("Numeros mayores al promedio");
25  ┌─ int cont =0;
26  ┌─ for(int i=0; i<num.length; i++){
27  ┌─     if(num[i]>prom){
28  ┌─         System.out.print(num[i]+", ");
29  ┌─     }else /*if(num[i]<prom || num[i] == prom)*/
30  ┌─         cont++;
31  ┌─     }
32  ┌─ }
33  ┌─ System.out.println();
34  ┌─ int [] num2 = new int[cont];
35  ┌─ int cont2 = 0;
36  ┌─ for (int i = 0; i < num.length; i++) {
37  ┌─     if(num[i]<prom || num[i] == prom){
38  ┌─         num2[cont2] = num[i];
39  ┌─         cont2++;
40  ┌─     }
41  ┌─ }
42  ┌─ System.out.println("Arreglo secundario con los numeros menores o iguales");
43  ┌─ System.out.println(Arrays.toString(num2));
44  ┌─
45  ┌─
46  ┌─ }
47  }
```



# UNIVERSIDAD TECNOLÓGICA DE ESCUINAPA

```
run:  
Ingresá el valor en la posición [0]  
10  
Ingresá el valor en la posición [1]  
10  
Ingresá el valor en la posición [2]  
10  
Ingresá el valor en la posición [3]  
9  
Ingresá el valor en la posición [4]  
9  
Ingresá el valor en la posición [5]  
8  
Ingresá el valor en la posición [6]  
7  
Ingresá el valor en la posición [7]  
6  
Ingresá el valor en la posición [8]  
7  
Ingresá el valor en la posición [9]  
8  
Promedio = 8  
Numeros mayores al promedio  
10, 10, 10, 9, 9,  
Arreglo secundario con los numeros menores o iguales  
[8, 7, 6, 7, 8]  
BUILD SUCCESSFUL (total time: 13 seconds)
```

16. Almacene en un arreglo de n números enteros (máximo 30) y determine cuántos números terminan en dígito primo.

```
public class A16 {  
    public static void main(String[] args) {  
        Scanner ent = new Scanner(System.in);  
        System.out.println("ingrese el tamaño del arreglo (Máximo 30)");  
        int n = ent.nextInt();  
        if(n>30){  
            System.out.println("Error: el tamaño no puede ser mayor a 30");  
            System.out.println("Ingrese otro número");  
            n = ent.nextInt();  
        }  
        int[] num = new int[n];  
        int cont = 0;  
        for(int i=0; i<num.length; i++){  
            System.out.println("Ingrese un número para la posición ["+i+"]");  
            num[i]=ent.nextInt();  
            int ultimod = num[i]%10;  
            if(ultimod == 2 || ultimod == 3 || ultimod == 5 || ultimod == 7){  
                cont++;  
            }  
        }  
        System.out.println("Cantidad de números que terminan en un número primo (2, 3, 5 y 7)\n"+cont);  
    }  
}  
  
run:  
ingrese el tamaño del arreglo (Máximo 30)  
4  
Ingrese un número para la posición [0]  
1  
Ingrese un número para la posición [1]  
2  
Ingrese un número para la posición [2]  
3  
Ingrese un número para la posición [3]  
4  
Cantidad de números que terminan en un número primo (2, 3, 5 y 7)  
2  
BUILD SUCCESSFUL (total time: 7 seconds)
```



# UNIVERSIDAD TECNOLÓGICA DE ESCUINAPA

17. Convierta un número entero decimal a su equivalente en representación binaria. (

```
11  */
12  public class A17 {
13      public static void main(String[] args) {
14          Scanner ent = new Scanner(System.in);
15          System.out.println("Ingresa un número para su conversión");
16          int n = ent.nextInt();
17          ArrayList<String> binario = new ArrayList<String>();
18          while (n != 0){
19              if(n%2 == 1){
20                  binario.add("1");
21              }else{
22                  binario.add("0");
23              }
24              n = n/2;
25          }
26      }
27      String num2 = "";
28      for(int i=binario.size()-1; i>=0; i--){
29          num2 += binario.get(i);
30      }
31      System.out.println("Número en binario "+num2);
32  }
33 }
```

```
Ingresar un número para su conversión
79
Número en binario 1001111
BUILD SUCCESSFUL (total time: 4 seconds)
```

18. Almacene en un arreglo n números enteros y determine ¿cuántas veces se repite cada uno de ellos?

Si  $n=6$  y los elementos del arreglo son: 3,4,6,6,4,6, se imprimirá:

3=1

4=2

6=3



# UNIVERSIDAD TECNOLÓGICA DE ESCUINAPA

```
10  /*
11  public class A18 {
12     public static void main(String[] args) {
13         Scanner ent = new Scanner(System.in);
14         System.out.println("Ingrese el tamaño del arreglo");
15         int n = ent.nextInt();
16         int[] num = new int[n];
17
18         for(int i=0; i<num.length; i++){
19             System.out.println("Ingresa un número para la posición ["+i+"]");
20             num[i]= ent.nextInt();
21         }
22         Arrays.sort(num);
23         int i = 0;
24
25         while (i < num.length) {
26             int numeroActual = num[i];
27             int contador = 1;
28
29             while (i + contador < num.length && num[i + contador] == numeroActual) {
30                 contador++;
31             }
32
33             System.out.println("El número " + numeroActual + " se repite " + contador + " vez/veces");
34             i += contador;
35         }
36     }
37 }
38 */
39
40
41
42 }
```

```
--
```

run:

Ingresar el tamaño del arreglo

4

Ingresar un número para la posición [0]

1

Ingresar un número para la posición [1]

2

Ingresar un número para la posición [2]

1

Ingresar un número para la posición [3]

1

El número 1 se repite 3 vez/veces

El número 2 se repite 1 vez/veces

BUILD SUCCESSFUL (total time: 8 seconds)

19. Almacene en un arreglo  $n$  números ( $n$  debe ser un numero par, y como máximo 50), valide si el tamaño es par e invertir los elementos a la mitad del arreglo. Imprimir el vector original y el invertido.

Si  $n=6$ ,  $v=[1][2][3][4][5][6]$   $v(\text{invertido})=[3][2][1][6][5][4]$ .



# UNIVERSIDAD TECNOLÓGICA DE ESCUINAPA

```
3  public class A19 {
4      public static void main(String[] args) {
5          Scanner ent = new Scanner(System.in);
6          System.out.println("Ingresa el tamaño de arreglo (solo números pares e igual o menor de 50)");
7          int n = ent.nextInt();
8          boolean v = true;
9          do{
10             if(n>50 || n%2 ==1){
11                 System.out.println("Error: el tamaño solo puede ser un número par y menor o igual que 50");
12                 System.out.println("Ingresa otro número");
13                 n = ent.nextInt();
14             }else{
15                 v = false;
16             }
17         }while(v);
18
19         int[] num = new int[n];
20
21         for(int i=0; i<num.length; i++){
22             System.out.println("Ingresa el número en la posición ["+i+"]");
23             num[i] = ent.nextInt();
24         }
25         System.out.print("v="+Arrays.toString(num));
26
27         int [] numInver = num.clone();
28         int mitad = n / 2;
29         for (int i = 0; i < mitad / 2; i++) {
30             int temp = numInver[i];
31             numInver[i] = numInver[mitad - 1 - i];
32             numInver[mitad - 1 - i] = temp;
33         }
34
35         for (int i = 0; i < mitad / 2; i++) {
36             int temp = numInver[mitad + i];
37             numInver[mitad + i] = numInver[n - 1 - i];
38             numInver[n - 1 - i] = temp;
39         }
40
41         System.out.println("  v(Invitido)= "+Arrays.toString(numInver));
42     }
43 }
44
run:
| Ingresa el tamaño de arreglo (solo números pares e igual o menor de 50)
4
| Ingresa el número en la posición [0]
1
| Ingresa el número en la posición [1]
2
| Ingresa el número en la posición [2]
3
| Ingresa el número en la posición [3]
4
| v=[1, 2, 3, 4]  v(Invitido)= [2, 1, 4, 3]
BUILD SUCCESSFUL (total time: 7 seconds)
```

20. Almacene en dos arreglos números reales a y b de 10 elementos cada uno, a partir de ellos genere un tercer arreglo con el siguiente criterio: sumar el primer elemento de a más el último elemento de b y luego el segundo elemento de a con el noveno elemento de b y así sucesivamente hasta llegar al décimo elemento de a más el primer elemento de b. Imprimir las sumas almacenadas en el arreglo c.



# UNIVERSIDAD TECNOLÓGICA DE ESCUINAPA

```
5 package arreglos2;
6 import java.util.*;
7 public class A20 {
8     public static void main(String[] args) {
9         Scanner ent = new Scanner(System.in);
10        int [] a = new int[10];
11        int [] b = new int[10];
12        int [] c = new int[10];
13
14        System.out.println("Llenado del arreglo a");
15        for(int i=0; i<a.length; i++){
16            System.out.println("Ingrese el valor en la posición ["+i+"]");
17            a[i] = ent.nextInt();
18        }
19
20        System.out.println("Llenado del arreglo b");
21        for(int i=0; i<b.length; i++){
22            System.out.println("Ingrese el valor en la posición ["+i+"]");
23            b[i] = ent.nextInt();
24        }
25
26        int cont1 = 0;
27        int cont2 = b.length-1;
28        while(cont1<a.length){
29            c[cont1] = a[cont1] + b[cont2];
30            cont1++;
31            cont2--;
32        }
33        System.out.println("Valores de a = "+Arrays.toString(a));
34        System.out.println("Valores de b = "+Arrays.toString(b));
35        System.out.println("Valores de c = "+Arrays.toString(c));
36    }
37}
38
```

Ingrese el valor en la posición [9]  
554  
Valores de a = [10, 1, 1, 1, 101, 101, 112, 110, 120, 11]  
Valores de b = [113, 32, 43, 43, 22, 12, 32, 42, 31, 554]  
Valores de c = [564, 32, 43, 33, 113, 123, 155, 153, 152, 124]  
BUILD SUCCESSFUL (total time: 33 seconds)

21. Almacene un arreglo de longitud n ordenado en forma ascendente y un elemento x del mismo tipo que los elementos del arreglo, intercalar x en el arreglo v de manera que siga ordenado.



# UNIVERSIDAD TECNOLÓGICA DE ESCUINAPA

```
11  public class A21 {
12      public static void main(String[] args) {
13          Scanner ent = new Scanner(System.in);
14          System.out.println("Ingrese el tamaño del arreglo");
15          int n = ent.nextInt();
16          ArrayList<Integer> num = new ArrayList<Integer>();
17          for(int i=0; i<n; i++){
18              System.out.println("Ingresa un valor para la posición ["+i+"]");
19              int c = ent.nextInt();
20              num.add(c);
21          }
22          num.sort((o1, o2) -> o1.compareTo(o2));
23          System.out.println("Arreglo ordenado");
24          for(int i=0; i<n; i++){
25              System.out.print(num.get(i)+", ");
26          }
27          System.out.println("\nIngrese un numero aleatorio");
28          int x = ent.nextInt();
29          num.add(x);
30          num.sort((o1, o2) -> o1.compareTo(o2));
31          System.out.println("Arreglo otra vez ");
32          for(int i=0; i<=n; i++){
33              System.out.print(num.get(i)+", ");
34          }
35      }
36  }

run:
Ingrese el tamaño del arreglo
2
Ingresa un valor para la posición [0]
1
Ingresa un valor para la posición [1]
2
Arreglo ordenado
1, 2,
Ingrese un numero aleatorio
0
Arreglo otra vez
0, 1, 2, BUILD SUCCESSFUL (total time: 16 seconds)
```

22. Almacene 2 arreglos a y b de longitudes n y m, el primero ordenado en forma ascendente y el segundo ordenado de manera descendente, crear un nuevo arreglo c de n + m elementos intercalando los elementos de a y b de modo que c quede ordenado en forma ascendente.



# UNIVERSIDAD TECNOLÓGICA DE ESCUINAPA

```
11  public class A22 {
12  public static void main(String[] args) {
13      Scanner ent = new Scanner(System.in);
14      System.out.println("Ingrese el tamaño del primer arreglo");
15      int n = ent.nextInt();
16      System.out.println("Ingrese el tamaño del segundo arreglo");
17      int m = ent.nextInt();
18
19      int[] a = new int[n];
20      Integer[] b = new Integer[m];
21
22      System.out.println("Llenado del arreglo a");
23      for(int i=0; i<a.length; i++ ){
24          System.out.println("Ingresa el valor en la posición ["+i+"]");
25          a[i] = ent.nextInt();
26      }
27      System.out.println("Llenado del arreglo b");
28      for(int i=0; i<b.length; i++ ){
29          System.out.println("Ingresa el valor en la posición ["+i+"]");
30          b[i] = ent.nextInt();
31      }
32      ArrayList <Integer> c = new ArrayList<Integer>();
33      for(int i=0; i<a.length; i++){
34          c.add(a[i]);
35      }
36      for(int i=0; i<b.length; i++){
37          c.add(b[i]);
38      }
39
40      Arrays.sort(a);
41      Arrays.sort(b, Collections.reverseOrder());
42      c.sort((o1, o2) -> o1.compareTo(o2));
43      System.out.println("arreglo a= "+Arrays.toString(a));
44      System.out.println("arreglo b= "+Arrays.toString(b));
45      System.out.print("arreglo c= [");
46      for(int i=0; i<n+m; i++){
47          System.out.print(c.get(i)+", ");
48      }
49      System.out.print("]");
```

```
run:
Ingresar el tamaño del primer arreglo
2
Ingresar el tamaño del segundo arreglo
3
Llenado del arreglo a
Ingresar el valor en la posición [0]
1
Ingresar el valor en la posición [1]
2
Llenado del arreglo b
Ingresar el valor en la posición [0]
2
Ingresar el valor en la posición [1]
3
Ingresar el valor en la posición [2]
4
arreglo a= [1, 2]
arreglo b= [4, 3, 2]
arreglo c= [1, 2, 2, 3, 4, ]BUILD SUCCESSFUL (total time: 13 seconds)
```

23. Calcular la suma de todos los elementos de un arreglo.



# UNIVERSIDAD TECNOLÓGICA DE ESCUINAPA

```
10  /*
11   *public class A23 {
12   *    public static void main(String[] args) {
13   *        Scanner ent = new Scanner(System.in);
14   *        System.out.println("Ingresa el tamaño del arreglo ");
15   *        int n = ent.nextInt();
16   *        int[] num = new int[n];
17   *
18   *        for(int i=0; i<n; i++){
19   *            System.out.println("Ingresa un valor para la posición ["+i+"]");
20   *            num[i] = ent.nextInt();
21   *        }
22   *        int suma =0;
23   *        for(int i=0; i<n; i++){
24   *            suma = suma + num[i];
25   *        }
26   *        System.out.println("Arreglo = "+Arrays.toString(num));
27   *        System.out.println("suma = "+suma);
28   *    }
29   */
30 }
```

```
run:
Ingresar el tamaño del arreglo
4
Ingresar un valor para la posición [0]
1
Ingresar un valor para la posición [1]
2
Ingresar un valor para la posición [2]
3
Ingresar un valor para la posición [3]
4
Arreglo = [1, 2, 3, 4]
suma = 10
BUILD SUCCESSFUL (total time: 7 seconds)
```

24. Realizar un arreglo que registre n palabras y muestre aquellas que inician con una vocal.

```
10  /*
11   *public class A24 {
12   *    public static void main(String[] args) {
13   *        Scanner ent = new Scanner(System.in);
14   *        System.out.println("Ingresa el tamaño del arreglo");
15   *        int n = ent.nextInt();
16   *        String [] p = new String[n];
17   *
18   *        for(int i=0; i<p.length; i++){
19   *            System.out.println("Ingresa la palabra para la posición ["+i+"]");
20   *            p[i] = ent.next();
21   *        }
22   *        String pvz = ""; //pvz de plantas vs zombies
23   *        for(int i=0; i<p.length; i++){
24   *            char l = p[i].charAt(0);
25   *            char lminuscula = Character.toLowerCase(l);
26   *            if(lminuscula == 'a' || lminuscula == 'e' || lminuscula == 'i' || lminuscula == 'o' || lminuscula == 'u'){
27   *                pvz += p[i]+", ";
28   *            }
29   *        }
30   *        System.out.println("palabras que inician con vocal \n"+pvz);
31   *    }
32   */
33 }
```



## UNIVERSIDAD TECNOLÓGICA DE ESCUINAPA

run:

```
Ingrresa el tamaño del arreglo
4
Ingresa la palabra para la posición [0]
arroz
Ingresa la palabra para la posición [1]
caballo
Ingresa la palabra para la posición [2]
JJBA
Ingresa la palabra para la posición [3]
infeliz
palabras que inician con vocal
arroz, infeliz,
```

**BUILD SUCCESSFUL (total time: 33 seconds)**

25. Escriba un arreglo que solicite las letras del alfabeto y las imprima de manera inversa.

```
10  /*
11   *public class A25 {
12   *     public static void main(String[] args) {
13   *         Scanner ent = new Scanner(System.in);
14   *         String[] alf = new String[27];
15   *         for(int i=0; i<alf.length; i++){
16   *             System.out.println("Ingrresa la letra numero "+(i+1)+" del alfabeto");
17   *             alf[i] = ent.next();
18   *         }
19   *         for(int i=alf.length-1; i>=0; i--){
20   *             System.out.println("letra["++(i+1)+"] = "+alf[i]);
21   *         }
22   *     }
23   */
24 }
```



## UNIVERSIDAD TECNOLÓGICA DE ESCUINAPA

Ingresá la letra numero 27 del alfabeto

ñ  
letra[27] = ñ  
letra[26] = z  
letra[25] = y  
letra[24] = x  
letra[23] = w  
letra[22] = v  
letra[21] = u  
letra[20] = t  
letra[19] = s  
letra[18] = r  
letra[17] = q  
letra[16] = p  
letra[15] = o  
letra[14] = n  
letra[13] = m  
letra[12] = l  
letra[11] = k  
letra[10] = j  
letra[9] = i  
letra[8] = h  
letra[7] = g  
letra[6] = f  
letra[5] = e  
letra[4] = d  
letra[3] = c  
letra[2] = b  
letra[1] = a

BUILD SUCCESSFUL (total time: 1 minute 34 seconds)

26. Una aplicación de las computadoras es dibujar gráficos de barra(llamados histogramas). Escriba un programa que dibuje un gráfico como el siguiente donde la cantidad de asteriscos de cada línea corresponda al número que está a la par.

3. | \*\*\*  
|  
5. | \*\*\*\*\*  
|  
7. | \*\*\*\*\*  
|  
9. | \*\*\*\*\*  
| \_\_\_\_\_



# UNIVERSIDAD TECNOLÓGICA DE ESCUINAPA

```
12  public class A26 {
13      public static void main(String[] args) {
14          int[] num = {3, 5, 7, 9};
15
16          for(int i=0; i<num.length; i++){
17              System.out.print(num[i]+". | ");
18
19              for (int j = 0; j < num[i]; j++) {
20                  System.out.print("*");
21              }
22
23              System.out.println();
24      }
25      System.out.print(" | ");
26      System.out.println("_".repeat(15));
27  }
```

Output x      Debugger Console x      ARREGLOS2 (run) x

```
run:
3. | ***
5. | *****
7. | ******
9. | *****

BUILD SUCCESSFUL (total time: 0 seconds)
```

27. Escribir un programa que visualice la siguiente salida:

```
1
1   2
1   2   3
1   2   3   4
1   2   3
1   2
1
```

```
12  public class A27 {
13      public static void main(String[] args) {
14
15          for (int i = 1; i <=4; i++) {
16
17              for (int j = 1; j <=i; j++) {
18
19                  System.out.print(j+"   ");
20              }
21              System.out.println();
22
23
24
25          for (int i = 3; i >=1; i--) {
26              for (int j =1; j <=i; j++) {
27                  System.out.print(j+"   ");
28              }
29              System.out.println();
30
31      }
32  }
```

Output x      Debugger Console x      ARREGLOS2 (run) x

```
run:
1
1   2
1   2   3
1   2   3   4
1   2   3
1   2
1

BUILD SUCCESSFUL (total time: 0 seconds)
```



## UNIVERSIDAD TECNOLÓGICA DE ESCUINAPA

28. Escriba un programa que imprima la forma en diamante siguiente.

```
*  
* *  
* * * *  
* * * * * *  
* * * * * * *  
* * * * * *  
* * * *  
* *  
*  
  
12  public class A28 {  
13      public static void main(String[] args) {  
14          int t = 3;  
15          for (int i = 0; i <= t; i++) {  
16              for (int es = 0; es < t - i; es++) {  
17                  System.out.print(" ");  
18              }  
19              for (int j = 0; j <= i * 2; j++) {  
20                  System.out.print("* ");  
21              }  
22              System.out.println();  
23          }  
24          for (int i = t - 1; i >= 0; i--) {  
25              for (int es = 0; es < t - i; es++) {  
26                  System.out.print(" ");  
27              }  
28              for (int j = 0; j <= i * 2; j++) {  
29                  System.out.print("* ");  
30              }  
31              System.out.println();  
32          }  
33      }  
34  }
```

A screenshot of an IDE interface. The code editor shows the Java code for generating a diamond pattern. Below it, the 'Output' tab displays the generated diamond pattern, followed by the message 'BUILD SUCCESSFUL (total time: 0 seconds)'.

29. Escriba un programa que imprima los siguientes patrones por separado, uno debajo del siguiente. Utilice ciclos “for” para generar los patrones.





# **UNIVERSIDAD TECNOLÓGICA DE ESCUINAPA**



# UNIVERSIDAD TECNOLÓGICA DE ESCUINAPA

30. Imprimir las siguientes series

$1 \times 9+2 = 11$ $12 \times 9+3 = 111$ $123 \times 9+4 = 1111$ $1234 \times 9+5 = 11111$ $12345 \times 9+6 = 111111$ $123456 \times 9+7 = 1111111$ $1234567 \times 9+8 = 11111111$ $12345678 \times 9+9 = 111111111$ $123456789 \times 9+10 = 1111111111$	$1$ $1 \times 1 = 1$ $11 \times 11 = 121$ $111 \times 111 = 12321$ $1111 \times 1111 = 1234321$ $11111 \times 11111 = 123454321$ $111111 \times 111111 = 12345654321$ $1111111 \times 1111111 = 1234567654321$ $11111111 \times 11111111 = 123456787654321$ $111111111 \times 111111111 = 12345678987654321$
--	---

```
12 public class A30 {
13     public static void main(String[] args) {
14
15         for (int i = 1; i < 10; i++) {
16             String numero = "";
17             for (int j = 1; j <= i; j++) {
18                 numero += j;
19             }
20
21             long num = Long.parseLong(numero);
22             long resultado = num * 9 + (i + 1);
23
24             System.out.println(numero + " × 9 + " + (i + 1) + " = " + resultado);
25         }
26
27         System.out.println("=".repeat(50));
28         for (int i = 1; i <= 10; i++) {
29             String numero = "1".repeat(i);
30
31             long num = Long.parseLong(numero);
32             long resultado = num * num;
33
34             System.out.println(numero + " × " + numero + " = " + resultado);
35         }
36     }
37 }
38 }
```

```
Debugger Console x  ARREGLOS2 (run) x
run:
1 × 9 + 2 = 11
12 × 9 + 3 = 111
123 × 9 + 4 = 1111
1234 × 9 + 5 = 11111
12345 × 9 + 6 = 111111
123456 × 9 + 7 = 1111111
1234567 × 9 + 8 = 11111111
12345678 × 9 + 9 = 111111111
123456789 × 9 + 10 = 1111111111
-----
1 × 1 = 1
11 × 11 = 121
111 × 111 = 12321
1111 × 1111 = 1234321
11111 × 11111 = 123454321
111111 × 111111 = 12345654321
1111111 × 1111111 = 1234567654321
11111111 × 11111111 = 123456787654321
111111111 × 111111111 = 1234567900987654321
BUILD SUCCESSFUL (total time: 0 seconds)
```