

Arreglos (arrays) o vectores

M.C. Ana Cristina Palacios García

3. Programa que declare un arreglo de diez elementos enteros y pida números para rellenarlo hasta que se llene el arreglo o se introduzca un número negativo. Entonces se debe imprimir el arreglo (sólo los elementos introducidos).

```

Algoritmo Ejercicio3
  //Declaramos el tamaño del arreglo x igual a 10
  Dimension x[10];

  //Debido a que la lectura de datos del vector está condicionada a detenerse cuando
  //se reciba un número negativo, y a asignar valores al arreglo en las celdas disponi-
  //bles mientras el número leído sea positivo o hasta que se llene el arreglo,
  //se propone resolver en primera instancia con el ciclo hacer-mientras (aunque tambien
  //es posible resolver con los otros ciclos)
  indice<-1; //esta variable llevará la cuenta de la posición en la que va a asignar
  //el siguiente valor leído, siempre y cuando sea positivo y esté dentro del intervalo
  //de valores de 1 a 10

  Repetir//Inicio del ciclo
    Escribir "Ingrese un valor entero: " Sin Saltar;
    Leer valor;

    si valor>=0 Entonces //Si el valor leído es positivo entonces
      x[indice]<-valor; //En el arreglo x en la posición que indica la variable
      //indice, se asigna el valor leído dado que fue positivo.
      indice<-indice+1; //se incrementa el valor del indice en 1, para indicar
      //el valor de la siguiente celda del arreglo en la que se podrá asignar el
      //siguiente valor leído siempre y cuando sea positivo.

    FinSi
    Hasta Que valor<0 o indice>10; //El ciclo se detiene cuando al menos una de estas
    //dos condiciones se cumple, es decir, se detiene si el valor leído es menor que
    //cero o negativo, o se detiene cuando el valor de la variable indice es superior a
    //10.

    //Mandamos a imprimir el arreglo, pero para imprimir solamente los elementos introduci-
    //dos, requerimos indicar que el ciclo inicie en 1 y termine hasta el valor que tiene
    //la variable indice menos 1
    Escribir "[" Sin Saltar;
    Para i<-1 Hasta (indice-1) Con Paso 1
      Escribir x[i],"," sin saltar;
    FinPara
    Escribir "]";
  FinAlgoritmo

```

```

*** Ejecución Iniciada. ***
Ingrese un valor entero: > 1
Ingrese un valor entero: > 2
Ingrese un valor entero: > 3
Ingrese un valor entero: > 4
Ingrese un valor entero: > -1
[1,2,3,4,]
*** Ejecución Finalizada. ***

```

```

*** Ejecución Iniciada. ***
Ingrese un valor entero: > 1
Ingrese un valor entero: > 2
Ingrese un valor entero: > 3
Ingrese un valor entero: > 4
Ingrese un valor entero: > 5
Ingrese un valor entero: > 6
Ingrese un valor entero: > 7
Ingrese un valor entero: > 8
Ingrese un valor entero: > 9
Ingrese un valor entero: > 10
[1,2,3,4,5,6,7,8,9,10,]
*** Ejecución Finalizada. ***

```

Prueba de funcionamiento en la que cuando se ingresa un valor negativo, el programa deja de solicitar datos e inmediatamente después, manda a imprimir el contenido del arreglo.

Prueba de funcionamiento en la que se demuestra que es posible leer los 10 varones enteros, y que dichos valores se asignan en cada celda del arreglo

```
package ejercicio3;
import java.util.Scanner;
public class Ejercicio3 {
    public static void main(String[] args) {
        int x[] = new int[10];
        Scanner sc=new Scanner(System.in);
        int indice=0;//Inicia en 0 dado que en Java los índices de los vectores comienzan en 0
        int valor=0;
        do{//Hacer desde aqui ...
            System.out.print("Ingrese un valor entero: ");
            valor=sc.nextInt(); //La instrucción para leer enteros en Java es nextInt()

            if(valor>=0){
                x[indice]=valor;
                indice=indice+1; //también es posible usar la instrucción indice++;
            }
        }while((valor>=0)&&(indice<10));//.. hasta aqui mientras que el valor sea positivo o mayor o igual
                                            //que cero y (&&) el valor del índice sea menor a 10
        //Mandamos a imprimir el arreglo
        System.out.print("[");
        for(int i=0; i<(indice); i++){
            System.out.print(x[i]+",");
        }
        System.out.println("]");
        System.out.println("El arreglo tiene: "+(indice)+" elementos");//No requerido en el problema
    }
}
```

Código Java para el ejercicio 3.

run:
Ingrese un valor entero: 1
Ingrese un valor entero: 2
Ingrese un valor entero: 3
Ingrese un valor entero: 4
Ingrese un valor entero: -1
[1,2,3,4,
El arreglo tiene: 4 elementos
BUILD SUCCESSFUL (total time: 5 seconds)

Prueba de funcionamiento

4. Crear un programa que pida un número al usuario un número de mes (por ejemplo, el 4 que equivale al mes de abril) y diga cuántos días tiene (por ejemplo, 30) y el nombre del mes. Debes usar un vector. Para simplificarlo vamos a suponer que febrero tiene 28 días.

Nota: Pseudocódigo y prueba de funcionamiento versión 1 de la solución
(recuerden que no hay una única forma de resolver el ejercicio).

```
1 Algoritmo Ejercicio4_version1
2 //Debido a que hay 12 meses, requerimos declarar un vector de tamaño 12
3 Dimension dias_mes[12]; //el vector se llama dias_mes
4 //Por cada mes (es decir, por cada celda o posición del arreglo), indicaremos la cantidad
5 //de días que tiene
6 dias_mes[1]=31; //el mes de enero se representa por la posición 1 en el arreglo dias_mes, y
7 | | | | | tiene 31 días
8 dias_mes[2]=28; //febrero
9 dias_mes[3]=31; //marzo
10 dias_mes[4]=30; //abril
11 dias_mes[5]=31;
12 dias_mes[6]=30;
13 dias_mes[7]=31; dias_mes[8]=31; //Agosto
14 dias_mes[9]=30; dias_mes[10]=31;
15 dias_mes[11]=30; dias_mes[12]=31; //diciembre
16
17 //Para que el programa diga también el mes, es posible tener un segundo arreglo con los nombres
18 //de los meses o usar una instrucción switch para filtrar los datos, vamos a usar el switch
19 Escribir "Ingrese un número del 1 al 12: " Sin Saltar;
20 Leer num;
21 Segun num Hacer
22
23 1: Escribir "Enero tiene ",dias_mes[num], " días";
24 2: Escribir "Febrero tiene ",dias_mes[num], " días";
25 3: Escribir "Marzo tiene ",dias_mes[num], " días";
26 4: Escribir "Abril tiene ",dias_mes[num], " días";
27 5: Escribir "Mayo tiene ",dias_mes[num], " días";
28 6: Escribir "Junio tiene ",dias_mes[num], " días";
29 7: Escribir "Julio tiene ",dias_mes[num], " días";
30 8: Escribir "Agosto tiene ",dias_mes[num], " días";
31 9: Escribir "Septiembre tiene ",dias_mes[num], " días";
32 10: Escribir "Octubre tiene ",dias_mes[num], " días";
33 11: Escribir "Noviembre tiene ",dias_mes[num], " días";
34 12: Escribir "Diciembre tiene ",dias_mes[num], " días";
35 De Otro Modo:
36 | | | | | Escribir "Error: Valor ingresado fuera del rango de 1 a 12..."; 
37 Fin Segun
38 FinAlgoritmo
```

```
1 package ejercicio4_version1;
2 import java.util.Scanner;
3 public class Ejercicio4_version1 {
4     public static void main(String[] args) {
5         Scanner sc = new Scanner(System.in);
6         int dias_mes[]={31,28,31,30,31,30,31,31,30,31,30,31}; //Declaramos el arreglo y lo inicializamos
7             //con la cantidad de días que tiene cada mes en orden ascendente, el primer valor de la
8             //izquierda corresponde al mes de enero, el segundo al de febrero, ...
9         System.out.print("Ingrese un número del 1 al 12: ");
10        int num = sc.nextInt(); //Leemos el número y lo asignamos a la variable num
11        switch(num){ //comparamos el valor del número con los siguientes casos y si coincide con alguno, se
12            //ejecuta el código de dicho caso, de lo contrario, se ejecuta el caso por default o defecto
13            case 1://Corresponde al primer mes del año, enero
14                System.out.println("Enero tiene: "+dias_mes[num-1]+" días");
15                break;
16            case 2://Corresponde al segundo mes del año, febrero
17                System.out.println("Febrero tiene: "+dias_mes[num-1]+" días");
18                break;
19            case 3://marzo
20                System.out.println("Marzo tiene: "+dias_mes[num-1]+" días");
21                break;
22            case 4://Abril
23                System.out.println("Abril tiene: "+dias_mes[num-1]+" días");
24                break;
25            case 5://Mayo
26                System.out.println("Mayo tiene: "+dias_mes[num-1]+" días");
27                break;
28            case 6://Junio
29                System.out.println("Junio tiene: "+dias_mes[num-1]+" días");
30                break;
31        }
32    }
33}
```

Código Java para el ejercicio 4, versión 1 (parte 1).

```
31     case 7://Julio
32         System.out.println("Julio tiene: "+dias_mes[num-1]+" días");
33         break;
34     case 8://Agosto
35         System.out.println("Agosto tiene: "+dias_mes[num-1]+" días");
36         break;
37     case 9://Septiembre
38         System.out.println("Septiembre tiene: "+dias_mes[num-1]+" días");
39         break;
40     case 10://Octubre
41         System.out.println("Octubre tiene: "+dias_mes[num-1]+" días");
42         break;
43     case 11://Noviembre
44         System.out.println("Noviembre tiene: "+dias_mes[num-1]+" días");
45         break;
46     case 12://Diciembre
47         System.out.println("Diciembre tiene: "+dias_mes[num-1]+" días");
48         break;
49     default:
50         System.out.println("Error: valor ingresado fuera del rango de 1 a 12... ");
51         break;
52     }
53 }
54 }
```

Código Java para el ejercicio 4, versión 1 (parte 2).

```
run:
Ingrese un número del 1 al 12: 5
Mayo tiene: 31 días
BUILD SUCCESSFUL (total time: 5 seconds)
```

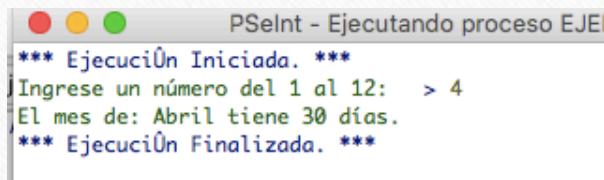
Prueba de funcionamiento cuando el número leído está dentro del rango [1,12].

```
run:
Ingrese un número del 1 al 12: 15
Error: valor ingresado fuera del rango de 1 a 12...
BUILD SUCCESSFUL (total time: 2 seconds)
```

Prueba de funcionamiento cuando el número leído está fuera del rango [1,12].

4. Crear un programa que pida un número al usuario un número de mes (por ejemplo, el 4 que equivale al mes de abril) y diga cuántos días tiene (por ejemplo, 30) y el nombre del mes. Debes usar un vector. Para simplificarlo vamos a suponer que febrero tiene 28 días.

Nota: Pseudocódigo y prueba de funcionamiento versión 2 de la solución (esta es la versión más adecuada para resolver el ejercicio).



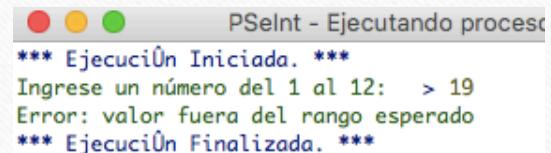
```
*** Ejecución Iniciada. ***
Ingresa un número del 1 al 12: > 4
El mes de: Abril tiene 30 días.
*** Ejecución Finalizada. ***
```

Prueba de funcionamiento 1: se proporciona un valor válido dentro del intervalo de 1 a 12

```
Algoritmo Ejercicio4_version2
    //Debido a que hay 12 meses, requerimos declarar un vector de tamaño 12
    Dimension dias_mes[12]; //el vector se llama dias_mes
    //Por cada mes (es decir, por cada celda o posición del arreglo), indicaremos la cantidad
    //de días que tiene
    dias_mes[1]=31; //el mes de enero se representa por la posición 1 en el arreglo dias_mes, y
    |           |           //tiene 31 días
    dias_mes[2]=28; //febrero
    dias_mes[3]=31; //marzo
    dias_mes[4]=30; //abril
    dias_mes[5]=31;
    dias_mes[6]=30;
    dias_mes[7]=31; dias_mes[8]=31; //Agosto
    dias_mes[9]=30; dias_mes[10]=31;
    dias_mes[11]=30; dias_mes[12]=31; //diciembre

    //Para que el programa diga también el mes, es posible tener un segundo arreglo con los nombres
    //de los meses
    Dimension nombre_mes[12];
    nombre_mes[1]="Enero"; nombre_mes[2]="Febrero";
    nombre_mes[3]="Marzo"; nombre_mes[4]="Abril";
    nombre_mes[5]="Mayo"; nombre_mes[6]="Junio";
    nombre_mes[7]="Julio"; nombre_mes[8]="Agosto";
    nombre_mes[9]="Septiembre"; nombre_mes[10]="Octubre";
    nombre_mes[11]="Noviembre"; nombre_mes[12]="Diciembre";

    Escribir "Ingrese un número del 1 al 12: " Sin Saltar;
    Leer num;
    //Para evitar errores, se requiere verificar que el valor proporcionado por el usuario está dentro del
    //intervalo de 1 a 12
    Si num=>1 y num<=12 Entonces //si el numero es mayor o igual a 1 y menor o igual a 12
        Escribir "El mes de: ",nombre_mes[num]," tiene ",dias_mes[num]," días.";
    SiNo
        Escribir "Error: valor fuera del rango esperado";
    FinSi
FinAlgoritmo
```



```
*** Ejecución Iniciada. ***
Ingresa un número del 1 al 12: > 19
Error: valor fuera del rango esperado
*** Ejecución Finalizada. ***
```

Prueba de funcionamiento 2: se proporciona un valor no válido fuera del intervalo de 1 a 12

```

1 package ejercicio4_version2;
2 import java.util.Scanner;
3 public class Ejercicio4_version2 {
4     public static void main(String[] args) {
5         Scanner sc = new Scanner(System.in);
6         int dias_mes[]={31,28,31,30,31,30,31,31,30,31,30,31}; //Declaramos el arreglo y lo inicializamos
7             //con la cantidad de días que tiene cada mes en orden ascendente, el primer valor de la
8             //izquierda corresponde al mes de enero, el segundo al de febrero, ...
9         String nombre_mes[]{"Enero","Febrero","Marzo","Abril","Mayo","Junio","Julio","Agosto",
10             "Septiembre","Octubre","Noviembre","Diciembre"}; //Declaramos e inicializamos
11             //el segundo arreglo que contiene los nombres de los meses
12         //Solicitamos al usuario el valor del mes
13         System.out.print("Ingrese un número del 1 al 12: ");
14         int num = sc.nextInt(); //Leemos el número y lo asignamos a la variable num
15
16         if(num>=1 && num<=12){ //Si num está dentro del intervalo [1,12], entonces podemos acceder a las celdas
17             //del arreglo, recordando restar un 1 al valor del número, dado que de lo contrario
18             //en un caso intentaremos acceder a la posición 12 del arreglo y esa no existe, dado
19             //que ambos arreglos inician en 0 y terminan en el indice 11 y tienen 12 elementos.
20             System.out.println("El mes de: "+nombre_mes[num-1]+" tiene "+dias_mes[num-1]+" días.");
21         }
22         else{//sino
23             System.out.println("Error: valor fuera del rango esperado");
24         }
25     }
26 }
27 }
```

Código de java del ejercicio 4 versión 2.

```

run:
Ingrese un número del 1 al 12: 12
El mes de: Diciembre tiene 31 días.
BUILD SUCCESSFUL (total time: 10 seconds)
```

Prueba de funcionamiento 1: se proporciona un valor válido dentro del intervalo de 1 a 12

```

run:
Ingrese un número del 1 al 12: 15
Error: valor fuera del rango esperado
BUILD SUCCESSFUL (total time: 2 seconds)
```

Prueba de funcionamiento 2: se proporciona un valor no válido fuera del intervalo de 1 a 12