Alessandra Gonzalez  
  
**Ejercicio 1:**

**Crear un arreglo de N elementos enteros. Llenar el arreglo con N números enteros provistos**

**por el usuario. Escribir una función que tome como parámetro el arreglo y devuelva la posición**

**del elemento más chico del arreglo. Imprimir por pantalla el menor de los elementos del**

**arreglo.**

import java.util.Scanner;

public class Main {

public static void main(String[] args) {

Scanner scanner = new Scanner(System.in);

// Solicitar al usuario el tamaño del arreglo

System.out.print("Ingrese la cantidad de elementos del arreglo: ");

int N = scanner.nextInt();

// Crear el arreglo de tamaño N

int[] arreglo = new int[N];

// Llenar el arreglo con números proporcionados por el usuario

System.out.println("Ingrese los elementos del arreglo:");

for (int i = 0; i < N; i++) {

System.out.print("Elemento " + (i + 1) + ": ");

arreglo[i] = scanner.nextInt();

}

// Obtener la posición del elemento más pequeño

int posicionMenor = obtenerPosicionMenor(arreglo);

// Imprimir el menor elemento del arreglo

System.out.println("El menor elemento del arreglo es: " + arreglo[posicionMenor] + " en la posición " + (posicionMenor + 1));

}

// Función para obtener la posición del elemento más pequeño en el arreglo

public static int obtenerPosicionMenor(int[] arreglo) {

int posicionMenor = 0;

for (int i = 1; i < arreglo.length; i++) {

if (arreglo[i] < arreglo[posicionMenor]) {

posicionMenor = i;

}

}

return posicionMenor;

}

}

**Ejercicio 2:**

**Crear un arreglo de N elementos enteros. Llenar el arreglo con N números enteros provistos**

**por el usuario. Escribir una función que tome como parámetro el arreglo y devuelva la posición**

**del elemento más grande del arreglo. Imprimir por pantalla el mayor de los elementos del**

**arreglo.**

import java.util.Scanner;

public class Main {

public static void main(String[] args) {

Scanner scanner = new Scanner(System.in);

// Solicitar al usuario el tamaño del arreglo

System.out.print("Ingrese la cantidad de elementos del arreglo: ");

int N = scanner.nextInt();

// Crear el arreglo de tamaño N

int[] arreglo = new int[N];

// Llenar el arreglo con números proporcionados por el usuario

System.out.println("Ingrese los elementos del arreglo:");

for (int i = 0; i < N; i++) {

System.out.print("Elemento " + (i + 1) + ": ");

arreglo[i] = scanner.nextInt();

}

// Obtener la posición del elemento más grande

int posicionMayor = obtenerPosicionMayor(arreglo);

// Imprimir el mayor elemento del arreglo

System.out.println("El mayor elemento del arreglo es: " + arreglo[posicionMayor] + " en la posición " + (posicionMayor + 1));

}

// Función para obtener la posición del elemento más grande en el arreglo

public static int obtenerPosicionMayor(int[] arreglo) {

int posicionMayor = 0;

for (int i = 1; i < arreglo.length; i++) {

if (arreglo[i] > arreglo[posicionMayor]) {

posicionMayor = i;

}

}

return posicionMayor;

}

**Ejercicio 3:**

**Crear un arreglo de N elementos enteros. Imprimir por pantalla el mensaje: “Todos positivos”**

**si todos sus elementos son positivos. En caso contrario imprimir por pantalla el mensaje: “No**

**todos positivos”.**

**Nota: se debería utilizar una única sentencia if en la impresión del mensaje por pantalla.**

import java.util.Scanner;

public class Main {

public static void main(String[] args) {

Scanner scanner = new Scanner(System.in);

// Solicitar al usuario el tamaño del arreglo

System.out.print("Ingrese la cantidad de elementos del arreglo: ");

int N = scanner.nextInt();

// Crear el arreglo de tamaño N

int[] arreglo = new int[N];

// Llenar el arreglo con números proporcionados por el usuario

System.out.println("Ingrese los elementos del arreglo:");

for (int i = 0; i < N; i++) {

System.out.print("Elemento " + (i + 1) + ": ");

arreglo[i] = scanner.nextInt();

}

// Verificar si todos los elementos son positivos

boolean todosPositivos = true;

for (int num : arreglo) {

if (num <= 0) {

todosPositivos = false;

break;

}

}

// Imprimir el mensaje correspondiente

if (todosPositivos) {

System.out.println("Todos positivos");

} else {

System.out.println("No todos son positivos");

}

}

}

**Ejercicio 4:**

**Crear un arreglo boléanos de N elementos. Calcular e imprimir por pantalla el producto de la**

**cantidad de los elementos que en una posición par del arreglo son true por la cantidad de**

**elementos que en posición impar son false.**

public class main  
{  
 public static void main(String [ ] args)  
 {  
 boolean vector [] = {true, false, false, true, true};  
 int producto = 0;  
 int cantTrue = 0;  
 int cantFalse = 0;

for (int i=0; i<vector.length; i++)

{  
 if (i%2 == 0)  
 {  
 if (vector [i] == true)  
 cantTrue = cantTrue + 1;

System.out.println(cantTrue);  
 }

else  
 {  
 if (vector[i] == false)  
 cantFalse = cantFalse + 1;  
 System.out.println(cantFalse);   
 }

}  
 producto = cantTrue \* cantFalse;  
 System.out.println(“El producto es: “ + producto);  
 }

}

**Ejercicio 5:**

**Crear dos arreglos de enteros de N elementos cada uno. Generar e imprimir por pantalla un**

**tercer arreglo que contenga los elementos de posiciones impares del primer arreglo y los de**

**posiciones pares del segundo arreglo.**

public class main  
{  
 public static void main(String [ ] args)  
 {  
 int vector1 []= {3,5,6,8,2,5,8,4};  
 int vector2 []= {1,4,5,7,3,1,2,9};  
 int vector3 []= new int[vector1.length];  
  
 for (int i=0; i<vector.length; i++)  
 {  
 if (i%2 != 0)  
 {

vector3[i] = vector1[i];

}  
 else  
 {  
 if (i%2 == 0)  
 {  
 vector3[i] = vector2[i];  
 }

}   
}

}  
}

**Ejercicio 6:**

**Crear dos arreglos de enteros de N elementos cada uno. Si el primer arreglo se referencia con la variable A y el segundo arreglo se referencia con la variable B. Generar e imprimir por pantalla un tercer arreglo en el cual sus elementos están dados por la siguiente fórmula: C(i) = A(i) \* B(N – i), si C es el nombre de la variable que referencia al tercer arreglo.**

public class Main

{

public static void main(String[] args) {

int numeros\_A[]={4,5,7,2};

int numeros\_B[]={6,3,8,1};

int numeros\_C[]= tercerVector\_C(numeros\_A,numeros\_B);

for(int i=0; i<numeros\_C.length; i++){

System.out.println("Posición "+"("+i+"): "+numeros\_C[i]);

}

}

public static int[] tercerVector\_C(int vector\_A[], int vector\_B[]){

int vectortem\_C[]={0,0,0,0};

for(int i=0; i<vectortem\_C.length; i++){

vectortem\_C[i]= (vector\_A[i] \* vector\_B[(vector\_B.length-1)-i]);

}

return vectortem\_C;

}

}

**Ejercicio 7:**

**Crear un arreglo de enteros de N elementos. Generar e imprimir por pantalla un arreglo que**

**contenga los elementos del primer arreglo en orden inverso.**

import java.util.Arrays;

import java.util.Random;

public class main  
{  
 public static void main(String [ ] args)  
 {  
 int vector[] = {0,2,6,7,8,5,9};  
  
 for (int i=vector.length -1; i>=0; i- -)  
 {  
 System.out.println(vector[i]);  
 }  
 }  
}

**Ejercicio 8:**

**Crear un arreglo de enteros de N elementos. Generar otro arreglo que es el resultado del**

**producto del arreglo por el escalar 3.14. El producto de un arreglo por un escalar se define**

**como la multiplicación de cada una de sus posiciones por el escalar. Imprimir por pantalla los elementos del arreglo resultante.**

public class Main

{

public static void main(String[] args)

{

int vector []= {3,5,6,8,2,5,8,4};

int[] arreglo = new int[vector.length];

for (int i = 0; i < vector.length; i++)

{

arreglo[i] = i + 1;

}

double[] arregloResultado = new double[vector.length];

for (int i = 0; i < vector.length; i++)

{

arregloResultado[i] = arreglo[i] \* 3.14;

}

System.out.println("Arreglo resultante:");

for (int i = 0; i < vector.length; i++)

{

System.out.println(arregloResultado[i]);

}

}

}

**Ejercicio 9:**

**Crear dos arreglos de enteros de N elementos. Calcular e imprimir por pantalla el resultado del producto escalar de los arreglos. El producto escalar de dos arreglos se define como la sumatoria del producto de los elementos que se encuentran en la misma posición en cada uno de los arreglos. Más formalmente, si llamamos A al primer arreglo y B al segundo:**

Dónde λ es el producto escalar.

public class Main

{

public static void main(String[] args) {

int vec\_num[]={4,7,9,4};

int vec\_num2[]={7,5,5,8};

int suma\_vectores=función\_escalar(vec\_num,vec\_num2);

System.out.println("El producto escalar de dos arreglos es: "+suma\_vectores);

}

public static int función\_escalar(int vec\_numeros[], int vec\_numeros2[]){

int vec\_temporal[]=new int[4];

int acumulador=0;

for(int i=0; i<vec\_numeros.length; i++){

vec\_temporal[i]=(vec\_numeros[i]\*vec\_numeros2[i]);

acumulador=(acumulador+vec\_temporal[i]);

}

return acumulador;

}

}

**Ejercicio 10:**

**Crear un arreglo de enteros de N elementos. Calcular e imprimir por pantalla el resultado del**

**producto escalar entre el arreglo y un arreglo que contenga los elementos del primer arreglo**

**en orden inverso.**

**Nota: no debería ser necesario duplicar el código para calcular el producto escalar ni el código**

**necesario para invertir los elementos de un arreglo. En ambos casos se deberían haber creado**

**métodos que realizaran dicha tarea.**

**Ejercicio 11:**

**Crear un arreglo de enteros de N elementos. Ingresar un número y buscarlo en el arreglo.**

**Imprimir por pantalla la posición del número si éste se encuentra en el arreglo. En caso**

**contrario imprimir por pantalla el mensaje: “El número buscado no se encuentra en el**

**arreglo”.**

import java.util.Scanner;

public class Main {

public static void main(String[] args) {

// Crear un arreglo de enteros de tamaño N

int[] arreglo = {5, 10, 15, 20, 25};

// Solicitar al usuario que ingrese un número a buscar

Scanner scanner = new Scanner(System.in);

System.out.print("Ingrese un número a buscar en el arreglo: ");

int numeroBuscado = scanner.nextInt();

// Buscar el número en el arreglo

boolean encontrado = false;

int posicion = -1;

for (int i = 0; i < arreglo.length; i++) {

if (arreglo[i] == numeroBuscado) {

encontrado = true;

posicion = i;

break;

}

}

// Imprimir el resultado de la búsqueda

if (encontrado) {

System.out.println("El número " + numeroBuscado + " se encuentra en la posición " + posicion + " del arreglo.");

} else {

System.out.println("El número buscado no se encuentra en el arreglo.");

}

}

}