

PROGRAMACIÓN JAVA EJERCICIOS-2 TEMA 5. Arrays Unidimensionales y métodos.

1. Programa que lea dos arrays V1 y V2 de 20 elementos numéricos enteros y a continuación crea un array V3 "concatenación" de los dos, es decir poner los elementos de V2 a continuación de los de V1. Mostrar los elementos de V3. Utiliza 3 métodos: uno para leer los arrays, otro para pasar los valores al array V3 y otro para mostrar los arrays.
2. Escribe un programa que lea un array de N elementos y rote todas sus componentes un lugar hacia su derecha, teniendo en cuenta que la última componente se ha de desplazar al primer lugar. No se puede utilizar un array auxiliar. Utiliza un método que realice el proceso de rotar las componentes del array.
3. Dado un número entero positivo de 6 cifras, escribe un programa que compruebe si el número es capicúa utilizando un array de 6 componentes. Utiliza un método para pasar las cifras del número al array y otro para calcular si es o no capicúa.
4. Escribe un método que reciba un array A de enteros y devuelva mediante return un array con los mismos elementos de A pero en orden inverso.
5. Escribe un método que reciba un array de enteros positivos que puede contener elementos duplicados y reemplace cada elemento repetido con el valor -1. El método devolverá el número de elementos modificados.
6. Se desea realizar un histograma con los pesos de los alumnos de un determinado curso:

| <u>Peso</u> | <u>Número de Alumnos</u> |
|-------------|--------------------------|
| 58 | ** |
| 62 | **** |
| 65 | ***** |
| 66 | ***** |
| ... | ... |

El número de asteriscos se corresponde con el número de alumnos del peso especificado. Realiza un programa que lea los pesos e imprima el histograma correspondiente. Suponer que los pesos están comprendidos entre los valores 45 y 100 Kg. En el histograma sólo aparecerán los pesos que se corresponden con 1 o más alumnos. Utiliza un método para leer los pesos y otro para mostrar el histograma.

7. Escribe un método que reciba un entero N y devuelva un array con N números aleatorios entre 0 y 9.
8. Escribe un método que reciba un entero N y devuelva un array con N números primos obtenidos de forma aleatoria.
9. Escribe un método que reciba dos arrays de tipo double y devuelva el mayor valor contenido en los dos arrays.
10. Escribe un método llamado *posicionNumero* que reciba tres parámetros: un array de enteros, un entero que representa una posición dentro del array y otro entero que es un número a buscar dentro del array. El método buscará el número dentro del array a partir de la posición indicada. El método devolverá la posición donde se ha encontrado el número o -1 si el número buscado no se encuentra en el array.