

**Ejercicios Tema 7. Estructuras de datos I. Strings y Arrays**

1. Escribe un método que dada una frase devuelva la misma pero con la primera letra en mayúsculas.
2. Escribe un programa que muestre el número de vocales y el número de consonantes de una frase.
3. Escribe un método que cuente el número de palabras que contiene. Asumimos que cada palabra está separada por solo un espacio en blanco.
4. Escribe un programa que muestre el número de vocales y el número de consonantes de cada palabra de una frase.
5. Escribe un programa que lea una frase y encuentre la palabra de mayor longitud. El programa debe imprimir la palabra y el número de caracteres que contiene.
6. Escribe un programa que lea una frase y un factor de multiplicación y devuelve la frase repetida tantas veces como indique el factor de multiplicación.
7. Escribe un programa que lea una frase y muestre cada palabra en una línea diferente.
8. Escribe un programa que dada una frase escriba en 2 columnas cada palabra seguida del número de caracteres que contiene.
9. Escribe un método que dada una frase devuelva otra pero con solo los caracteres de las posiciones impares.
10. Escribe un método que devuelva true si una palabra es palíndroma (se escribe igual de izquierda a derecha que de derecha a izquierda) y false si no lo es.
11. Escribe un un programa que solicite tres cadenas de caracteres, una que se denomine nombre, la otra primerCognom y la otra segonCognom. El programa debe crear una cadena con el nombre completo y después mostrar por pantalla lo siguiente:
  - a) El nombre completo con todos el caracteres en minúscula, después en mayúscula y después su longitud.
  - b) Los 5 primeros caracteres de la cadena (solo en el caso de que la longitud sea de 5 o más caracteres).
  - c) Los dos últimos caracteres de la cadena (solo en el caso de que la longitud sea de dos o más caracteres).
  - d) El número de ocurrencias en la cadena del último carácter.

- e) La cadena con todas las ocurrencias del primer carácter en mayúscula
- f) La cadena con tres \* por delante y por detrás.
- g) La cadena invertida.

12. Escribe un método que, dada una cadena de caracteres, sustituya todas las ocurrencias del texto «es» por «no por». Escribe un segundo método que sustituya todos el grupos de dígitos por un único carácter \*, es decir, si la cadena de caracteres es «esto1234es5678bueno900», el primer método tiene que devolver «no porto1234no por5678bueno900» y el segundo tiene que devolver «esto\*es\*bueno\*».

Nota:

- a) La expresión regular `\\d+` significa un dígito o más.
- b) Podemos usar `replaceAll("\\d", "*");`

Podéis encontrar más información sobre expresiones regulares en Java a:

<http://docs.oracle.com/javase/7/docs/api/java/util/regex/pattern.html#sum>

[http://chuwiki.chuidiang.org/index.php?title=expresiones\\_regulares\\_en\\_java](http://chuwiki.chuidiang.org/index.php?title=expresiones_regulares_en_java)

- 13. Escribe un programa para introducir los valores de un array de 10 elementos numéricos y posteriormente visualizar-los. Visualizar también la media de los elementos.
- 14. Escribe un programa que permita introducir un array de 10 elementos de tipos char y después visualizar los que ocupan la posición par. Repetir para las posiciones impar.
- 15. Escribe un programa que permita introducir un array de 20 elementos numéricos y visualizarlos de 4 en 4 en una tabla tabulada.
- 16. Escribe un programa que permita generar una secuencia de 30 números aleatorios de tipos double entre 0 y 10 (imagina que son notas de alumnos) y los almacene en un array. Calcula e imprime la media de aquellos elementos mayores o iguales que 5 y también la media total.
- 17. Escribe un programa que permita introducir una secuencia de N (pediremos N al usuario) estaturas de personas y guardarlas en un array. Calcular la media de todas ellas y mostrar información de cuántas de ellas tienen estatura superior a la media.
- 18. Genera dos arrays de 10 elementos de tipo entero entre 1 y 10. Después genera otro array suma que sea el resultado de la suma de los elementos de los 2 arrays. Genera también otro array donde el valor de sus elementos sea la división de los elementos de los dos arrays iniciales.

Consideraciones:

Crea métodos a los que le pasaremos los dos arrays iniciales y devuelvan otro con el resultado (suma o división).

19. Genera un array de 100 enteros entre 0 y 100 (aleatorios) y crea 3 métodos que impriman los que son múltiplos de 5, los que son 0 y los que son múltiplos de 10.
20. Genera un método al que le pasaremos un array y un número y nos devuelva si ese número está en el array.
21. Implementa los siguientes métodos (respeta las cabeceras que se proporcionan):
  - a) `public static void crearArray(int[] vector):` rellena el array con números aleatorios del 0 al 50.
  - b) `public static void visualitzarArray(int[] vector):` visualiza todo el contenido del array.
  - c) `public static void visualitzarParell(int[] vector):` muestra la posición y el contenido de los elementos que tienen valor par.
  - d) `public static void visualitzarMultiple3(int[] vector):` muestra la posición y el contenido de los elementos que son múltiplo de 3.

El array (de 10 elementos) se debe declarar, crear e inicializar en el main. Además, se hará un menú parecido al siguiente, desde el cual se invocarán a los métodos creados:

MENÚ PRINCIPAL

=====

1.-Rellenar array.  
 2.-Visualizar contenido del array  
 3.-Visualizar contenido par.  
 4.-Visualizar contenido múltiplo de 3  
 0.-Salir del menú.  
 Selecciona una opción:

22. Escribe un programa que permita introducir un array P de 10 elementos numéricos, visualizar su contenido y crear otro array S con los elementos de P que sean mayores de 10 (cuando los elementos sean  $\leq 10$ , se pondrá un -1). Visualizar después el contenido de S.
23. Escribe un programa que permita, a partir de un array de 10 elementos numéricos V, generar un array P con los elementos de V en orden inverso. Visualizar los dos arrays al final.
24. Escribe un programa que permita, a partir de un array V de 50 elementos, generar un array P de 50 elementos de forma que:
 
$$P(0) = V(0)$$

$$P(1) = V(0) + V(1)$$

$$P(2) = V(0) + V(1) + V(2)$$

....

$$P(49) = V(0) + V(1) + V(2) + \dots + V(49)$$

25. Escribe un programa que permita, a partir de un array V de 50 elementos, crear e imprimir un array P con los elementos de V de contenido par. Si no hubiera, que lo indique mediante un mensaje.
26. Crea una matriz de 4 filas y 8 columnas de enteros (aleatorios) y a continuación visualiza:
- Toda la matriz (por filas)
  - La matriz traspuesta (por columnas)
  - La matriz por filas con la suma total de cada fila
  - Los valores máximo y mínimo de la matriz
  - La media de toda la matriz
  - El elemento que nos indique el usuario (Dime la fila? Dime la columna? )
  - La fila que nos indique el usuario
  - La columna que nos indique el usuario
  - Guarda en la columna 8 el total de todas las anteriores.
27. Crea un vector V numérico de 50 elementos y otro P de 20 elementos. A continuación genera M(50,20) de tal forma que M(i,j) es igual a V(i)\*P(j).
28. Crea un array de 500 letras mayúsculas aleatorias y después, con un método, obtener una tabla similar a la siguiente donde el número representa el número a veces que se repite cada letra.
- A -->25
- B -->10
- .....
- Z -->12
- SUGERENCIA: Para transformar un entero en una letra: (char)numAleatori
29. Crea un programa para visualizar las notas de un grupo escolar y mostrar algunas estadísticas. El grupo se compone de 20 alumnos y 3 módulos (asignaturas).
- Para hacer la entrada de datos más rápida, generaremos las notas de los módulos aleatoriamente. Las estadísticas que tenemos que calcular son las siguientes:
- Calcular la media de cada alumno
  - La máxima nota de cada módulo
  - La nota media de cada módulo y cuántos alumnos la sobrepasan.

El programa debe mostrar las notas y las estadísticas de la siguiente forma:

	<u>MOD 1</u>	<u>MOD 2</u>	<u>MOD 3</u>	<u>MEDIA</u>
ALUMNO 1	5.00	3.32	6.56	4.96
ALUMNO 2	8.00	8.42	7.22	7.88
.....				
ALUMNO 20	6.33	4.55	7.21	6.03
NOTA MÁXIMA	8.00	8.42	7.22	
MEDIA (nalum)	6.22 (7)	5.76 (7)	6.33 (9)	