

Programación

UD05 – Tipos de Datos Compuestos

Ejercicios 1

Recuerda que siempre es una buena idea diseñar el algoritmo que resuelva el problema antes de empezar a programar y representarlo bien con un diagrama de flujo o bien con pseudocódigo.

Arrays.

1. Crea un array con los valores 15, 12, 8, 9 y 50.
2. Crea un array con los valores 5, 1.0, -8, -9.5 y 3.14.
3. Crea un array con los valores 'H', 'o', 'l' y 'a'.
4. Crea un array con los valores "variables", "literales", "constantes", "expresiones", "métodos", "bloques" y "excepciones".
5. Crea un programa que genere un array de 10 números aleatorios mayores o iguales a 0 y menores que 100. Después el programa mostrará los valores por pantalla en una sola línea separados por espacios. A continuación, entrará en bucle indefinido y pedirá al usuario que introduzca un número entero, el programa indicará si dicho número está o no en el array. El programa parará cuando el usuario introduzca un número negativo.
6. Crea un programa que genere un array de 10 números aleatorios entre -10 y 10. Después el programa mostrará los valores por pantalla en una sola línea separados por espacios. A continuación, mostrará cuántos de los números son negativos.
7. Crea un programa que genere un array de 10 números aleatorios entre -10 y 10. Después el programa mostrará los valores por pantalla en una sola línea separados por espacios. A continuación, debe cambiar los números negativos del array por el mismo valor positivo. El programa mostrará el array de nuevo.

8. Crea un programa que genere dos arrays de 10 números aleatorios entre -10 y 10. El programa mostrará los valores de cada array. A continuación, el programa creará un nuevo array de 10 números cuyos valores sean la suma de los elementos con el mismo índice de los arrays obtenidos previamente. Después se mostrarán los números almacenados en el array resultante.
9. Crea un programa que genere un array de 10 números aleatorios entre -10 y 10. El programa mostrará los números contenidos en el array. Después el programa debe mostrar los elementos que se repiten.
10. Crea un programa que genere un array de 10 números aleatorios entre -10 y 10. El programa mostrará los números contenidos en el array. Después el programa debe mostrar los elementos que no se repiten.
11. Crea un programa que genere un array de 10 números aleatorios entre -10 y 10 de manera que no haya elementos repetidos. El programa mostrará los números contenidos en el array.
12. Crea un programa (o clase) java llamado IntArray
 - a) Codifica una función que genere un array de una longitud dada de enteros aleatorios pertenecientes a un rango dado empezando en un entero mínimo dado.

```
public static int[] createRandom(int length, int min, int bound);
```

- b) Codifica una función que recorra todos los elementos de un array de enteros y los muestre por consola en una línea.

```
public static void showArray(int[] a);
```

- c) Codifica una función que busque un valor entre los elementos de un array de enteros, el valor de retorno de la función será el primer índice en el que se ha encontrado al número o -1 si no existe en el array.

```
public static int find(int[] a, int search);
```

- d) Codifica una función que busque un valor entre los elementos de un array de enteros a partir de un índice dado, el valor de retorno de la función será el primer índice en el que se ha encontrado al número o -1 si no existe en el array.

```
public static int find(int[] a, int search, int fromIndex);
```

- e) Codifica una función que devuelva el mínimo de los datos en el array.

```
public static int getMin(int[] a);
```

- f) Codifica una función que devuelva el máximo de los datos en el array.

```
public static int getMax(int[] a);
```

- g) Codifica una función que devuelva la suma de los datos en el array.

```
public static int getSum(int[] a);
```

- h) Codifica una función que devuelva la media aritmética de los datos en el array.

```
public static double getAvg(int[] a);
```

- i) Si queremos obtener los datos que devuelven las funciones anteriores estamos recorriendo el array 4 veces. Esto no es un problema para pocos valores, pero sería un desperdicio de tiempo para arrays con muchos datos. Codifica una función que reciba un array de tipo int y devuelva otro array de longitud 4 y tipo double con los siguientes valores y en ese orden: suma, media, mínimo, máximo.

```
double[] getStats(double[] a);
```

13. Crea un programa que pida al usuario introducir una lista de nombres; primero pedirá cuantos nombres va a introducir, después pedirá introducir esos nombres. Los datos se guardarán en un array. Después el programa mostrará la lista ordenada por orden alfabético.
14. Pueblos de Aragón. El programa mostrará al usuario las siguientes opciones:
- Mostrar pueblos por provincia. Pedirá al usuario que elija la provincia: Huesca, Teruel, Zaragoza. Después mostrará todos los pueblos de la provincia ordenados por orden alfabético.
 - Introducir pueblos por provincia. Pedirá al usuario que elija la provincia: Huesca, Teruel, Zaragoza. Después pedirá al usuario el número de pueblos de la provincia elegida que va a introducir y, a continuación, pedirá que los introduzca. Una vez introducidos añadirá los nuevos pueblos a la lista de pueblos de la provincia de seleccionada validando que no hay duplicados.

- c) Mostrar todos los pueblos de las tres provincias en orden alfabético.
- d) Buscar pueblo. Buscará en los arrays de las provincias si existe el pueblo buscado. Si existe mostrará el resultado y a qué provincia pertenece. Si no lo encuentra ofrecerá al usuario introducirlo pidiéndole para ello la provincia a la que pertenece.