

PROGRAMACIÓN EN VISUAL BASIC

Ejercicios sin resolver

Se recomienda codificar la solución en papel antes de pasarlo al ordenador.

No se deben usar variables globales, ni la instrucción GOTO.

CONCEPTOS BÁSICOS

1. Pedir al usuario dos números, uno entero y otro real, y mostrar su producto.
2. Calcular el área de un círculo cuyo radio se le preguntará al usuario.
3. Pedir al usuario los coeficientes (a , b , c) de un polinomio de segundo grado (ax^2+bx+c) y mostrar las dos raíces: $(-b \pm (b^2-4ac)^{1/2}) / 2a$.
4. Preguntar al usuario su nombre (texto) y su edad (valor entero) y mostrar por pantalla un texto del estilo del siguiente: *Te llamas Lucía y tienes 19 años.*
5. Pedir un entero al usuario y mostrar su valor al cuadrado. Probar el programa con el valor 30.000. En caso de que el programa falle, razonar por qué ocurre.
6. Preguntar una cantidad en euros e indicar la cantidad de monedas de cada tipo que se necesitarán para entregar esa cantidad. Se utilizarán las monedas de mayor valor posible.

Por ejemplo. $7,91\text{€} = 3 \times 2\text{€} + 1 \times 1\text{€} + 1 \times 0,50\text{€} + 2 \times 0,20\text{€} + 1 \times 0,01\text{€}$

ESTRUCTURAS DE CONTROL: DECISIONES

7. Pedir un número entero al usuario e indicar si es par o impar.
8. Preguntar a un usuario cuántos años tiene y, si es mayor de edad, preguntarle si tiene carné de conducir.
9. Pedir al usuario tres valores reales de doble precisión y mostrar el menor de ellos.
10. Repetir el ejercicio 3, teniendo en cuenta las posibles raíces complejas.
11. Pedir tres números reales al usuario y mostrar el menor o el mayor, según decida.
12. Escribir un programa (usando la instrucción *IF*) que pida un número real de precisión sencilla y haga muestre los siguientes resultados:
 - Si el número es menor que 50, mostrará su cuadrado
 - Si el número es igual a 25, 30 ó 75, mostrará su raíz cúbica
 - Si el número vale 10 ó es mayor que 100 ó está en los intervalos $[3,8]$ o $[77, 90]$, mostrará el valor dividido por 10.
 - En cualquier otro caso, mostrará el número leído.
13. Sabiendo que los descuentos de una tienda son del 5% cuando se compra más de 300 €, del 10% cuando se compra más de 500 € y del 12% para cantidades mayores de 800€, escribe un programa que pregunte al usuario la cantidad comprada y le indique el importe a pagar. Usar la función *IF*.
14. Repetir el ejercicio 12 usando la instrucción *SELECT CASE*.
15. Repetir el ejercicio 13 usando la instrucción *SELECT CASE*.

ESTRUCTURAS DE CONTROL: BUCLES

16. Pedir un valor N al usuario y calcular la serie $1^2+2^2+3^2\dots$ hasta que la suma iguale o supere el valor N .
17. Preguntar un valor par al usuario. Se repetirá la pregunta mientras el número introducido sea impar.
18. Pedir dos valores al usuario: $N (> 1000)$ y K (entre 2 y 10). Se mostrará cuántas veces es N divisible por K .
19. Escribe un programa que calcule el factorial de un número solicitado al usuario.
20. Haz un programa que pregunte a un niño la tabla de multiplicar (del 2 al 9). Si la respuesta es incorrecta se indicará el resultado correcto y en caso contrario se le felicitará.
21. Crear un programa que solicite al usuario un valor entero e indique si dicho valor es primo o no.
22. Dados dos números, calcular todos los número primos intermedios.
23. Calcular el máximo que alcanza el polinomio $x^5-15x^4+49x^3+111x^2-482x+336$ en el intervalo $x \in [1,7]$ con una precisión de 0,001
24. Calcular el valor de la constante matemática e con la siguiente serie:
$$e = 1/0! + 1/1! + 1/2! + 1/3! + \dots$$

El cálculo se detendrá cuando se hayan acumulado tantos términos de la serie como indique el usuario, para lo cual se le preguntará un valor N al principio del programa.
25. Repetir el programa anterior, pero deteniendo el programa cuando la diferencia entre los dos últimos términos calculados sea menor que un valor d solicitado al usuario al principio del programa.
26. Se escribirá un programa que permita jugar al *Master Mind*. Para ello se escribirá el programa de la siguiente forma:
 - Generar cuatro valores enteros aleatorios entre 0 y 9 distintos entre si.
 - Preguntar al usuario por cuatro valores enteros.
 - Considerando ambas combinaciones, indicar cuántos números comunes están en la misma posición y cuántos valores comunes están en distinta posición.
Ejemplo: 3786 y 8760 → uno colocado (el 7) y dos descolocados (8 y 6)
 - Se terminará el programa cuando el jugador acierte la combinación o haya agotado sus intentos (constante MAXINTENTOS con valor 10)

ESTRUCTURAS DE DATOS: MATRICES

27. Pedir al usuario el número de filas y de columnas de una matriz bidimensional y asignar a cada posición el valor resultante de sumar la fila y la columna en la que se encuentra.
28. Pedir una matriz al usuario y sumar todos los elementos que se encuentren en una fila o columna impar.
29. Obtener la suma de todos los elementos sobre la diagonal principal de una matriz solicitada al usuario.
30. Preguntar al usuario una lista de números y calcular la media y su desviación típica.
31. Solicitar al usuario una matriz y mostrar las posiciones y los valores del máximo y el mínimo.

32. Preguntar al usuario el grado y los coeficientes de un polinomio. Preguntar también por un valor real y evaluar el polinomio en ese valor.
33. Escribir un programa que calcule la suma de dos matrices cuyas dimensiones y valores se le preguntarán al usuario.
34. Escribir un programa que calcule el producto de dos matrices cuyas dimensiones y valores se le preguntarán al usuario.
35. Ampliar el ejercicio 24 de forma que al concluir la partida, se muestre todos las jugadas del usuario.
36. Dada una matriz bidimensional $n \times n$, crear otra de las mismas dimensiones, donde el valor de cada elemento, sea el promedio del elemento en la misma posición de la primera matriz y de los vecinos que lo rodean (nueve o menos).

Ejemplo

El elemento 2ª fila, 2ª columna es el promedio de todos sus vecinos (marcados con doble línea)

matriz inicial					matriz resultado				
1	3	6	2	3
2	2	4	4	4
7	9	2
...
...

ESTRUCTURAS DE DATOS: REGISTROS

37. Declarar un registro que permita guardar el nombre del titular de una cuenta bancaria, el número de la cuenta, el DNI del titular y el saldo. Declarar tres variables de este tipo, pedirle al usuario los valores y mostrar por pantalla cada uno de ellos. Finalmente, se mostrará la suma de los tres saldos.
38. Se pedirá al usuario los nombres y las notas de los alumnos de una clase y se almacenará en el correspondiente vector de registros. Se calculará la nota media, el porcentaje de aprobados y se indicará el nombre del alumno con la máxima nota.
39. Almacenar en un vector un conjunto de números complejos solicitados al usuario. Mostrar la suma de todos.
40. Definir un registro que permita almacenar el nombre y el teléfono de una persona. Crear un vector de K registros y mostrar un menú que permita añadir personas a la agenda, así como buscar teléfonos. K se declarará como constante.

FUNCIONES Y PROCEDIMIENTOS

NOTAS:

- *las funciones no leerán información por teclado (InputBox) ni mostrarán por pantalla ningún mensaje (MsgBox).*
 - *En cada ejercicio, se debe escribir un programa principal que permita comprobar las funciones y procedimientos de este apartado.*
41. Modificación del ejercicio 2: escribir una función que devuelva (sin mostrarlo por pantalla) el área de un círculo cuyo radio se le suministra como argumento.
 42. Modificación del ejercicio 3: suministrar como argumentos a una función los tres coeficientes de una ecuación de segundo grado, y devolver una de sus raíces.
 43. Crear un procedimiento que reciba un valor como argumento y lo cambie de signo.
 44. Crear un procedimiento que reciba dos variables reales como argumentos e intercambiar sus valores.
 45. Modificación del ejercicio 9: pasar tres argumentos reales a una función que devolverá el menor de ellos.
 46. Modificación del ejercicio 13: crear una función reciba como argumento el importe de una compra y devuelva la cantidad final a pagar, teniendo en cuenta que los descuentos son del 5% cuando se compra más de 300 €, del 10% cuando se compra más de 500 € y del 12% para cantidades mayores de 500€, escribe un programa que pregunte al usuario la cantidad comprada y le indique el importe a pagar. Usar preferiblemente la función *SELECT*.
 47. Crear una función que reciba un entero como argumento y devuelva un valor lógico que indique si el número es primo o no.
 48. Modificación del ejercicio 24: escribir una función que devuelva el valor de la constante matemática e como la serie $e = 1/0! + 1/1! + 1/2! + 1/3! + \dots$
El programa se detendrá cuando se hayan acumulado tantos términos de la serie como indique el único argumento que se se pasará a la función.
 49. Modificación del ejercicio 26: Se modificará el programa del juego del Master Mind, escribiendo y usando las siguientes funciones:
 - Función que recibe como argumento un vector de cuatro posiciones enteras y devuelve un valor lógico que indicará si contiene valores repetidos (*True*) o no (*False*).
 - Procedimiento que reciba un vector de cuatro posiciones y devuelva en cada una de ellas un número aleatorio entre 0 y 9. Los cuatro valores han de ser diferentes.
 - Función que reciba dos vectores de cuatro posiciones y devuelva un entero que indicará cuantos valores coinciden en la misma posición en ambos vectores.
 - Función que reciba dos vectores de cuatro posiciones y devuelva un entero que indicará cuantos valores coinciden en ambos vectores pero en posiciones distintas.
 50. Modificar el ejercicio 41, usando dos funciones: la primera buscará el teléfono asociado al nombre y la segunda añadirá un nombre y un teléfono al vector, siempre que queden posiciones libres. Se recuerda que no se deben usar variables globales en ninguno de los ejercicios.

MÓDULOS

51. Modificar el ejercicio 50 guardando el procedimiento *Main* en un módulo y el resto de las rutinas en otro.
52. Modificar el problema 51, guardando las declaraciones del registro y de la constante en un módulo, las funciones en otro y el procedimiento *Main* en un tercer módulo.
53. Escribir un módulo que contenga la declaración de un registro con dos enteros que representarán en número de filas y columnas de una matriz bidimensional y un tercer campo que será la matriz propiamente dicha. En un segundo módulo, se escribirán dos funciones que devolverán la multiplicación y la suma de dos matrices. Finalmente, se escribirá en un tercer módulo el procedimiento *Main* que pedirá dos matrices al usuario (con el número de columnas y filas que él decida) y mostrará su suma y su producto.

FICHEROS DE ACCESO ALEATORIO

Nota: no se puede cargar el contenido de los ficheros en vectores.

54. Solicitar de forma repetida al usuario la información de un modelo de automóvil (marca, modelo y precio) y guardarlo en un fichero de acceso aleatorio. Tras cada escritura se le preguntará al usuario si quiere seguir introduciendo más datos.
55. Sobre el fichero del ejercicio anterior, realizar las siguientes operaciones.
 - a. Búsqueda del modelo más barato y del más caro.
 - b. Preguntar por un modelo y una marca al usuario y buscarlo. Si se localiza, preguntar al usuario si quiere cambiarle el precio y actuar en consecuencia. Si no se encuentra, indicarlo con un mensaje por pantalla.
 - c. Calcular el precio medio de los vehículos.
56. Ampliar el ejercicio anterior añadiendo una cuarta opción para permitir al usuario consultar precios de vehículos.
57. Modificar el ejercicio 54 para que antes de introducir la información de un nuevo vehículo se compruebe que no está ya en el archivo. Si ya existe, le dará la posibilidad al usuario de modificar el precio.
58. Escribir un programa que permita almacenar ventas de distintos representantes comerciales. Para ello se guardará en cada posición de un fichero de acceso aleatorio el nombre del vendedor, la fecha (vector de tres enteros: día, mes y año) y el importe de esa venta. El programa mostrará un menú al usuario y le ofrecerá las siguientes opciones:
 - a. Introducir una nueva venta.
 - b. Modificar el importe de una venta: se solicitará el nombre del vendedor y la fecha de la venta. Si no la encuentra avisará al usuario por medio de un mensaje en pantalla.
 - c. Buscar todas las ventas de un vendedor.
 - d. Mostrar el importe total de ventas de cada vendedor.
 - e. Solicitar un año y mostrar el importe total de ventas en ese año.
 - f. Mostrar el peor vendedor de un año concreto (solicitar el año al usuario)
 - g. Salir del programa.

Tras ejecutar la acción solicitada, se volverá a mostrar el menú.

FICHEROS DE TEXTO

Nota: no se puede cargar el contenido de los ficheros en vectores.

59. Preguntar al usuario la marca, modelo y unidades de los automóviles que quiere adquirir. Guardar esta información en un fichero de texto.
60. Usando el fichero del ejercicio anterior y el del ejercicio 54, calcular el importe de la compra.
61. Ampliar el ejercicio anterior de modo que se genere un fichero de texto con una línea de información para cada modelo de coche que se quiere comprar: marca, modelo, unidades y gasto total en ese modelo (unidades adquiridas x precio por unidad).
62. Este ejercicio utilizará dos ficheros de texto. En el primero se almacenará el nombre y DNI de los alumnos de una clase (un alumno por línea). En el segundo se guardarán las notas que el alumno va obteniendo a lo largo del curso. En cada línea contendrá el DNI, el nombre de la asignatura, la convocatoria (mes) y la nota (puede tener decimales). Este programa mostrará un menú con las siguientes opciones:
 - a. Introducir un nuevo alumno en el primer fichero descrito: nombre y DNI.
 - b. Introducir una nueva nota en el segundo fichero: DNI, asignatura y nota.
 - c. Solicitar el nombre de un alumno y mostrar todas sus notas.
 - d. Solicitar el nombre de un alumno y una asignatura e indicar si ha aprobado la asignatura o no y cuántas convocatorias ha consumido.
 - e. Solicitar al usuario una asignatura y una convocatoria y mostrar el número de examinados, el de aprobados así como su porcentaje.
 - f. Solicitar al usuario una asignatura y una convocatoria y mostrar el nombre del alumno con mejor nota.

Notas:

- No se puede suponer ningún tipo de orden en los ficheros. Por ejemplo, la nota de junio de un alumno puede aparecer después de la de septiembre.
 - Si un alumno no se presenta a un examen, no aparecerá su nota (de esa convocatoria) en el fichero de notas.
63. Modificar el apartado c ejercicio anterior para que, además de mostrar por pantalla las notas, se guarden en un fichero de texto (el nombre del fichero será el DNI del alumno y extensión *txt*).
 64. Modificar el ejercicio anterior de modo que el segundo fichero descrito (el de notas) sea de acceso aleatorio.