

Objetivos:

- ❖ Diseñar **diagramas de flujo** con **estructuras repetitivas**.
- ❖ Codificar programas VB con **sentencias repetitivas** tipo **While** y **Do-Loop**.
- ❖ Realizar programas **combinando** sentencias **condicionales** y sentencias **repetitivas**.

Programa demostración de sentencias repetitivas While y Do-Loop

Interfaz

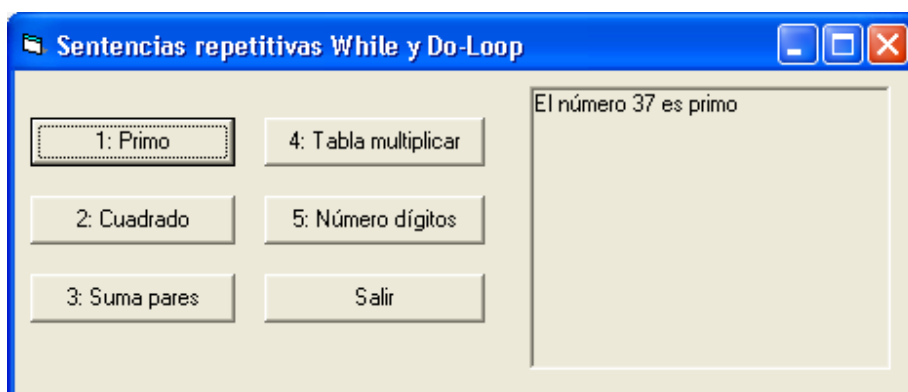


Figura 6.1 Objetos presentes en la interfaz: botones y caja de dibujo

Funcionamiento

1. Cada **ejercicio** tendrá un botón de ejecución propio (**cmdEj1**, **cmdEj2**, ..., **cmdEj5**).
2. Al hacer **click** sobre cada botón, borraremos inicialmente el contenido de la **caja de dibujo** (PictureBox) del resultado, **pctRes**. Utilizaremos para ello el método **cls** (**pctRes.cls**).
3. Al hacer **click** en el botón **Salir**, el **programa finalizará**.
4. Se proporciona un modelo de programa **ejecutable** para clarificar los enunciados.

Ejercicio 6.1: decir si un número positivo es primo¹ (resolución)

Enunciado

Diseña el diagrama de flujo y codifica un programa que lea un número positivo y diga si es primo.

¹ Número natural que tiene dos divisores naturales **distintos**: él mismo y el 1. Por definición **el 1 no es primo**.

Funcionamiento

Cuando el usuario pulse el botón etiquetado “1: Primo”, el programa pedirá un número positivo mediante una instrucción **InputBox** y mostrará en la caja de dibujo si ese número es **primo**, tal y como se ejemplifica en la Figura 6.1.

Algoritmo

Hay muchos algoritmos para saber si un número positivo **n** es primo. El que se propone aquí consiste en buscar el primer número **d** que partiendo de 2 sea divisor de **n**. Si **d** llega a **n** quiere decir que el número es primo. Este algoritmo es válido para números mayores que 1. En la Figura 6.2 se muestra el diagrama de flujo de un programa para leer un número mayor que 1 y verificar si es primo.

Diagrama de flujo

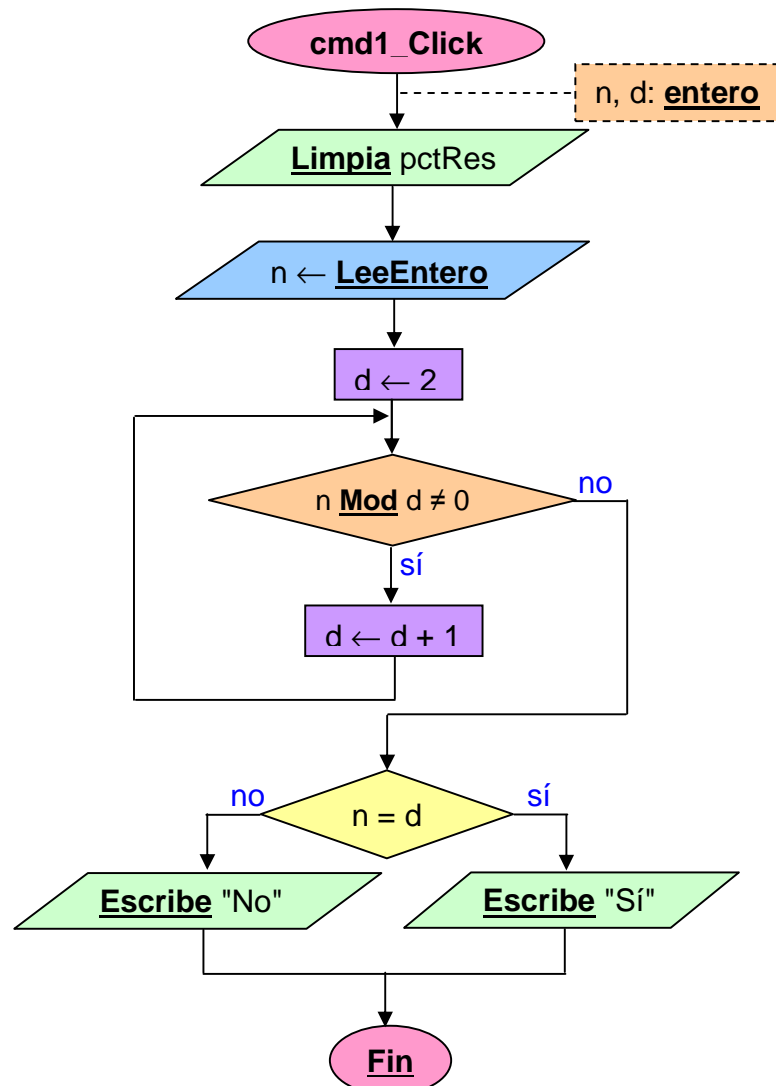


Figura 6.2 Diagrama de flujo del ejercicio 1 para decir si un número es primo

Pasos a seguir

1. Crearemos los objetos del tipo y forma mostrados en la Figura 6.1. Guardaremos todo.
2. Añadir el código a los eventos, es decir, el clic sobre los botones. En nuestro ejemplo resuelto, al hacer clic sobre el primer botón se ejecutará el código de la Figura 6.3.

```
Sub cmdEj1_Click()  
  Dim s As String  
  Dim n As Integer, d As Integer  
  
  pctRes.Cls  
  s = InputBox("Introduce un número natural positivo")  
  n = CInt (s)  
  d = 2  
  While n Mod d <> 0  
    d = d + 1  
  Wend  
  If n = d Then  
    pctRes.Print "El número " & n & " es primo"  
  Else  
    pctRes.Print "El número " & n & " no es primo"  
  End If  
End Sub
```

Figura 6.3 Código VB para escribir si un número es primo.

Ejercicios propuestos

1. **Codifica** el programa (mayormente resuelto) que dice si un número es primo **verificando** que el dato introducido sea **numérico** y que sea un número **positivo**. Añade el caso específico del 1 que no es primo. Para otros no-primos especifica el primer divisor encontrado.
2. **Diseña** el diagrama de flujo y **codifica** un programa que lea una secuencia de números finalizada en 0 y vaya mostrando en el cuadro de dibujo el cuadrado de cada uno de los números excepto el último 0.
3. **Diseña** el diagrama de flujo y **codifica** un programa que lea una secuencia de números naturales finalizada en 0 y vaya mostrando en el cuadro de dibujo la suma de los números **pares**.
4. Diseña el **diagrama de flujo** y escribe el **programa** que pida y lea repetidas veces un número **n** (utilizando **InputBox**) y nos muestre la **tabla de multiplicar** del 1 al 9 de ese número **n** hasta leer el 0.
5. Diseña el **diagrama de flujo** y escribe el **programa** que pida un número **natural** y calcule la cantidad de **dígitos en base decimal** que tiene ese número. Para poder utilizar números mayores utiliza un tipo Double.

Tablas de referencia rápida

Sintaxis	Ejemplo	Diagrama de flujo
<u>While</u> <i>cond</i> ... <u>Wend</u>	<u>While</u> $p < n$ $p = p * 10$ <u>Wend</u>	
<u>Do</u> ... <u>Loop While</u> <i>cond</i>	<u>Do</u> $s = s + 1$ $n = n / 10$ <u>Loop While</u> $n \geq 10$	
<u>Do</u> ... <u>Loop Until</u> <i>cond</i>	<u>Do</u> $s = s + 1$ $n = n / 10$ <u>Loop Until</u> $n < 10$	
<u>For</u> $i = ini$ <u>To</u> fin <u>Step</u> <i>paso</i> ... <u>Next</u> i	<u>For</u> $i = 1$ <u>To</u> 5 <u>Step</u> 1 $pctRes.Print\ i$ <u>Next</u> i	