



Objetivos:

- ❖ Diseñar **diagramas de flujo** con **estructuras repetitivas**.
- ❖ Codificar programas VB con **sentencias repetitivas** tipo **For**.
- ❖ Realizar programas **combinando** sentencias **condicionales** y sentencias **repetitivas**.
- ❖ Tipo **Long**

Programa de demostración del uso de sentencias For

Interfaz

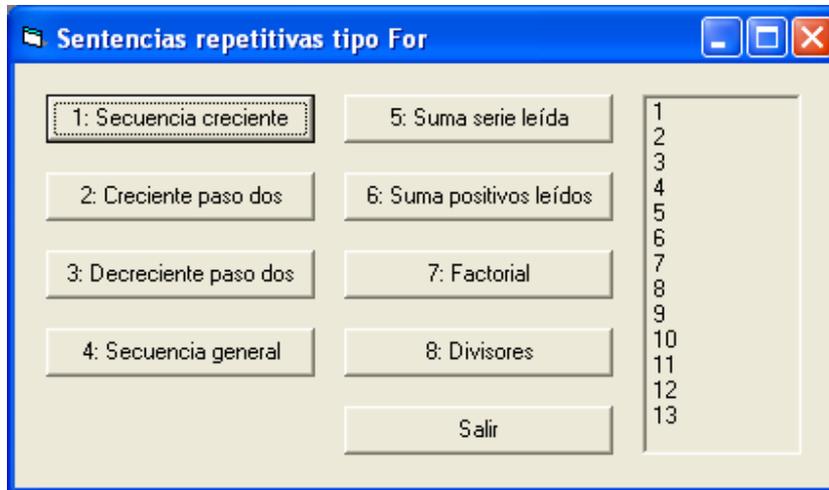


Figura 5.1 Objetos presentes en la interfaz: botones y caja de dibujo

Funcionamiento general

1. Cada **ejercicio** tendrá un botón de ejecución propio (**cmdEj1**, **cmdEj2**, ..., **cmdEj8**).
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 5.1: escribir secuencia incremental de números (resolución)

Funcionamiento

Cuando el usuario pulse el botón etiquetado “1: Secuencia creciente”, el programa pedirá el límite máximo y mostrará en la caja de dibujo la secuencia de números del 1 al máximo proporcionado, de uno en uno, cada uno en una línea distinta, tal y como se muestra en la Figura 5.1 para un límite de 13.

Diagrama de flujo

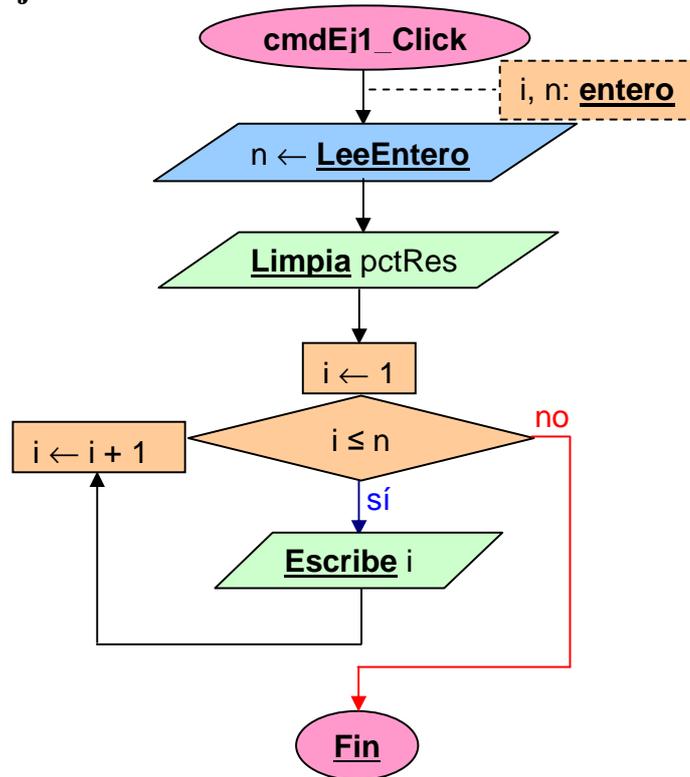


Figura 5.2. Diagrama de flujo del ejercicio 1 utilizando For.

Pasos a seguir

1. Crearemos los objetos del tipo y forma mostrados en la Figura 5.1. Guardaremos todo.
2. Añadir el código a los eventos. Por ejemplo, al hacer clic sobre el botón “1: Secuencia creciente” el código asociado será el de la figura 5.3.

```

Sub cmdEj1_Click()
    Dim s As String
    Dim i As Integer, n As Integer
    s = InputBox("Número de elementos")
    n = Cint(s)
    pctRes.Cls
    For i = 1 To n Step 1
        pctRes.Print i
    Next i
End Sub
  
```

Figura 5.3 Código VB para escribir la secuencia creciente de paso 1.

Ejercicios propuestos

Resuelto el código del primero de los botones se propone la resolución del resto. Sólo se verificará la validez de los datos de entrada si se pide explícitamente.

2. **Diseña** el diagrama de flujo y **codifica** el programa VB que pida un número **máximo** y muestre en el cuadro de dibujo una **secuencia creciente de dos en dos, comenzando por el dos** hasta este número leído: 2, 4, 6, 8, 10, ..., **máximo**.
3. **Diseña** el diagrama de flujo y **codifica** el programa VB que pida un número **máximo** y muestre la **secuencia de números positivos decreciente de paso 2** que comienza en ese número: **máximo**, ..., 9, 7, 5, 3, 1. **Verifíquese** que el dato leído es **numérico** y **positivo**.
4. **Codifica** el programa VB que lea un número de partida **ini**, un número tope **fin** y un paso **pas** y nos muestre la serie que comienza en **ini** y termina en **fin**. Si **pas** es positivo la secuencia será **creciente** y si es negativo será **decreciente**. Nótese que es una llamada directa a la instrucción **For** en VB por lo que si no hay ningún número entre **ini** y **pas** (y son distintos) no nos mostrará ningún número. Este comportamiento no se puede representar de modo genérico en el diagrama de flujo ya que la condición de parada tendrá un signo \leq en un caso y \geq en el otro.
5. **Diseña** el diagrama de flujo y **codifica** el programa VB que pida el número de elementos a sumar **n** y los vaya pidiendo (r_1, r_2, \dots, r_n), mostrando el resultado de la **suma** final $r_1+r_2+r_3+\dots+r_n$. El tipo de los elementos será **Double**.
6. **Diseña** el diagrama de flujo y **codifica** el programa VB que lea el número de elementos a sumar **n** y a continuación los vaya pidiendo, mostrando el resultado de la **suma** final $r_1+r_2+r_3+\dots+r_n$ de aquéllos que sean positivos, **ignorando los negativos**. El tipo de los elementos será **Double**.
7. **Diseña** el diagrama de flujo y **codifica** el programa VB que pida un número natural **n** y escriba su factorial, **n!**. Prueba los siguientes casos: $0! = 1$, $1! = 1$, $4! = 24$, $9! = 362880$. Se recomienda utilizar enteros largos (**Long**). Obsérvese que a partir del factorial de 13 habrá desbordamiento.
8. **Diseña** el diagrama de flujo y **codifica** el programa VB que pida un número natural **n** y nos muestre el número y todos sus divisores a partir del 2.

Tablas de referencia rápida

Sintaxis	Ejemplo	Diagrama de flujo
<pre>For i= ini To fin Step paso ... Next i</pre>	<pre>For i=1 To 5 Step 1 pctRes.Print i Next i</pre>	<pre> graph TD Start(()) --> Init[i ← ini] Init --> Cond{i ≤ fin} Cond -- Sí --> Inc[i ← i + paso] Inc --> Cond Cond -- No --> Exit(()) </pre>