



1. (1 punto) Completa los huecos de la siguiente tabla con las equivalencias en las respectivas bases, utilizando los dígitos necesarios en cada caso. **Incluye todas las operaciones que hayas necesitado para hacerlo.** Nota: se representan solamente **números positivos**.

Base 2	Base 4	Base 7	Base 10	Base 16
111010				
	111			
		111		

2. (2 puntos) Una organización de protección civil recibe información constante de la velocidad del viento, queriendo organizar los protocolos de actuación ante huracanes según la escala de Saffir-Simpson, que los clasifica de la siguiente manera:

Velocidad (km/h)	119-153	154-177	178-209	210-249	A partir de 250
Categoría	1	2	3	4	5
Daños	Mínimos	Moderados	Extensos	Extremos	Catastróficos

- Escribe un subprograma VB que reciba la velocidad del viento en km/h y devuelva la categoría del huracán ó 0 si no se puede considerar como tal.
- Escribe un subprograma VB que dada la categoría de un huracán escriba en pantalla un mensaje escueto de los daños que se pueden esperar.

3. (3 puntos) Podemos definir una serie geométrica a partir del primer elemento a_0 y la razón r de manera que cada elemento a_i para $i > 0$ se calcula a partir del anterior $a_i = r \cdot a_{i-1}$. Ejemplo:

$a_0 = 4$, $r = 3$, $a_1 = 4 \cdot 3 = 12$, $a_2 = 12 \cdot 3 = 36$, $a_3 = 36 \cdot 3 = 108$, ...

Codifica un programa VB que pida el primer elemento a_0 (un número natural) y la razón r (otro número natural, $r > 1$) de una serie geométrica y luego pida un número natural x para mostrar en pantalla si pertenece o no a la serie. Verifica la corrección de los datos, si son numéricos y si cumplen las restricciones expresadas, volviendo a pedirlos en caso de que no sean correctos.

Variable leída	Valor introducido	Control	Mensaje de error (si lo hay)
a_0	dos	Nok	No es numérico
a_0	-1	Nok	No es natural
a_0	0	Ok	
r	B	Nok	No es numérico
r	1	Nok	No es mayor que 1
r	4	Nok	
x	hola	Nok	No es numérico



4. (4 puntos) **Diseña** (el diagrama de flujo) y **codifica** una **función** VB que reciba dos números naturales en base 10 **como cadenas** de tamaño arbitrario (por ejemplo de 100 dígitos) y los sume, devolviendo la cadena del resultado. Téngase en cuenta que no vale convertir estos números a enteros ya que no funcionaría para números grandes.

Escribe un programa de prueba que lea dos números naturales y los sume.

Como apunte algorítmico se recomienda invertir ambas cadenas e ir sumando los dígitos de menor peso a mayor peso. Supóngase que se dispone de una función que invierte una cadena con el siguiente prototipo:

Invertir (ByVal cad As String) As String	Invierte la cadena cad, p. ej. "Hola" a "aloH"
---	--

Ejemplo: Sumar "3457" y "149128393084" (hágase primero a mano).

```

{s1, s2} = {"3457" y "149128393084"}
Invertimos s1 y s2 a i1 e i2:
{i1, i2, llevo} = {"7543" y "480393821041", 0}
7+4+0= 11 → 1 y llevo 1 → "1" y degrado las cadenas i1, i2
{i1, i2, llevo, res} = {"543" y "80393821041", 1, "1"}
5+8+1= 14 → 4 y llevo 1 → "41"
{i1, i2, llevo, res} = {"43" y "0393821041", 1, "41"}
4+0+1= 5 → 5 y llevo 0
{i1, i2, llevo, res} = {"3" y "393821041", 0, "541"}
3+3+0= 6 → 6 y llevo 0
{i1, i2, llevo, res} = {"" y "93821041", 0, "6541"}
0+9+0= 9 → 9 y llevo 0
{i1, i2, llevo, res} = {"" y "3821041", 0, "96541"}
Y así sucesivamente

```

Pista: para obtener el valor numérico de un dígito, por ejemplo, "8" se recomienda utilizar la función **Val**.

Tabla de referencia de Visual Basic	
IsNumeric (ByVal cad As String) As Boolean	Verifica si una cadena contiene un valor numérico
Val (ByVal cad As String) As Integer	Valor numérico de una cadena
Mid (ByVal cad As String, ByVal ini As Long [, ByVal lon As Long]) As String	Subcadena desde <i>ini</i> hasta la longitud <i>lon</i> indicada, o hasta el final de la cadena si no se indica
Len (ByVal cad As String) As Integer	Longitud de una cadena