

## **FUNCION PARA EL CALCULO DEL DIGITO VERIFICADOR EN PL/SQL**

```
CREATE OR REPLACE FUNCTION Pa_Calcular_Dv_11_A (  
    p_numero    IN VARCHAR2,  
    p_base_max  IN NUMBER DEFAULT 11)  
    RETURN NUMBER IS  
/*  
    Calcula Dígito Verificador numérico con entrada alfanumérica y base_max 11  
*/  
v_total      NUMBER(6);  
v_resto      NUMBER(2);  
k            NUMBER(2);  
v_numero_aux NUMBER(1);  
v_numero_al  VARCHAR2(255);  
v_caracter   VARCHAR2(1);  
v_digit      NUMBER;  
BEGIN  
    -- Cambia la última letra por ASCII en caso que la cédula termine en letra  
    FOR i IN 1 .. LENGTH(p_numero) LOOP  
        v_caracter := UPPER(SUBSTR(p_numero,i,1));  
        IF ASCII(v_caracter) NOT BETWEEN 48 AND 57 THEN -- de 0 a 9  
            v_numero_al := v_numero_al || ASCII(v_caracter);  
        ELSE  
            v_numero_al := v_numero_al || v_caracter;  
        END IF;  
  
    END LOOP;  
    -- Calcula el DV  
    k := 2;  
    v_total := 0;  
  
    FOR i IN REVERSE 1 .. LENGTH(v_numero_al) LOOP  
        IF k > p_base_max THEN  
            k := 2;  
        END IF;  
        v_numero_aux := TO_NUMBER(SUBSTR(v_numero_al,i,1));  
        v_total := v_total + (v_numero_aux * k);  
        k := k + 1;  
    END LOOP;  
  
    v_resto := MOD(v_total,11);  
  
    IF v_resto > 1 THEN  
        v_digit := 11 - v_resto;  
    ELSE  
        v_digit := 0;  
    END IF;  
  
    RETURN v_digit;  
  
END;
```

## **FUNCION PARA EL CALCULO DEL DIGITO VERIFICADOR EN VISUAL BASIC**

Function getDV(RUC As String) As String

```
getDV = calcular(RUC, 11)
End Function
```

```
Function calcular(numero As String, basemax As Integer) As String
Dim codigo As Long
Dim numero_al As String
```

```
Dim i
For i = 1 To Len(numero)
    Dim c
    c = Mid$(numero, i, 1)
    codigo = Asc(UCase(c))
    If Not (codigo >= 48 And codigo <= 57) Then
        numero_al = numero_al & codigo
    Else
        numero_al = numero_al & c
    End If
Next
```

```
Dim k: Dim total
k = 2
total = 0
```

```
For i = Len(numero_al) To 1 Step -1
    If (k > basemax) Then k = 2
    Dim numero_aux
    numero_aux = Val(Mid(numero_al, i, 1))
    total = total + (numero_aux * k)
    k = k + 1
Next
```

```
Dim resto: Dim digito
resto = total Mod 11
If (resto > 1) Then
    digito = 11 - resto
Else
    digito = 0
End If
calcular = digito
End Function
```

## **FUNCION PARA EL CALCULO DEL DIGITO VERIFICADOR EN LENGUAJE C**

```
#include
#include
#include
```

```
int Pa_Calcular_Dv_11_A (const char *p_numero, int p_basemax)
{
    // CALCULO Digito verificador - RUC
    // AUTOR: Luis Francou.
```

```

int v_total, v_resto, k, v_numero_aux, v_digit;
char *v_numero_al, *p;
char v_caracter;
int i;

v_numero_al = (char *) malloc (strlen(p_numero) + 1);
memset(v_numero_al, 0, strlen(p_numero)+1);

while (v_caracter = *p_numero++, v_caracter != 0)
{
    if (v_caracter >= '0' && v_caracter <= '9')
        sprintf(v_numero_al, "%s%c", v_numero_al, v_caracter);
    else {
        sprintf(v_numero_al, "%s%d", v_numero_al, v_caracter);
    }
}

k = 2;
v_total = 0;
i = strlen(v_numero_al) - 1 ;
p = v_numero_al + i++;

while (i--)
{
    k = k > p_basemax ? 2 : k;
    v_numero_aux = *(p--) - 48;
    v_total += v_numero_aux * k++;
}

v_resto = v_total % 11;
v_digit = v_resto > 1 ? 11 - v_resto : 0;

free(v_numero_al);

return v_digit;
}

```