

1._Algoritmo de descomposición de un número en sus dígitos, del menos significativo al más significativo:

El proceso consiste en determinar el dígito como el residuo de dividir el número por 10 y repetir el proceso con un nuevo número formado por la parte entera de la división por 10, mientras existan dígitos.

Por ejemplo: Considere el número: **45897**.

NUMERO	DIGITO OBTENIDO	NUEVO NUMERO	CÓDIGO
45897	$45897 \% 10 = 7$	$45897 // 10 = 4589$	while N != 0: dig = N % 10 N = N // 10
4589	$4589 \% 10 = 9$	$4589 // 10 = 458$	
458	$458 \% 10 = 8$	$458 // 10 = 45$	
45	$45 \% 10 = 5$	$45 // 10 = 4$	
4	$4 \% 10 = 4$	$4 // 10 = 0$	

Note que si N es la variable que guarda el número cuyos dígitos vamos a descomponer al finalizar el proceso el valor de esta variable es cero, por lo tanto se recomienda utilizar una variable auxiliar donde se almacene el valor de N, también podemos notar que este algoritmo descompone el número desde el dígito menos significativo al dígito más significativo: **7 9 8 5 4**.

Supongamos que queremos hacer una aplicación en Python que lea un número entero y determine e imprima la suma de sus dígitos:

```
# Inicializamos variables
num = 0 # Variable con el numero a descomponer
dig = 0 # Variable que guarda su dígito
suma = 0 # Variable que almacena la suma de los dígitos
temp = 0 # Variable temporal
# Entrada del dato
num = int(input("De un número entero: "))
# Ciclo que descompone el número y suma sus dígitos
temp = num
while temp != 0: # El ciclo se ejecuta hasta temp = 0
    dig = temp % 10 # Obtenemos dígito menos significativo
    suma += dig # Acumulamos dígito en suma
    temp //= 10 # Eliminamos el dígito obtenido de temp
print("")
print("****Salida Resultados****")
print("Para el número: ", num, " la suma de sus dígitos es: ", suma)
print("\nFin del programa")
```

Al ejecutar el programa tendríamos la siguiente salida por consola:

```
De un número entero: 23456
Para el número: : 23456 la suma de suma dígitos es 20
```

2._Descomposición de un número en sus dígitos, del más significativo al menos significativo

El proceso consiste en determinar el dígito como el cociente de dividir el número por la unidad seguida de tantos ceros como la posición del dígito dentro del número original y repetir el proceso con un nuevo número formado de quitar el dígito obtenido, mientras existan dígitos.

Por ejemplo: Considere el numero: **45897**.

NUMERO	DIGITO OBTENIDO	NUEVO NUMERO	PROCESO
45897	$45897 // 10000 = 4$	$45897 \% 10000 = 5897$	1. Determinar la Potencia para obtener el primer dígito (P) 2. Determinar cada dígito for i in range(1, nroDigitos +1): dig = N // P N = N % P P = P // 10
5897	$5897 // 1000 = 5$	$5897 \% 1000 = 897$	
897	$897 // 100 = 8$	$897 \% 100 = 97$	
97	$97 // 10 = 9$	$97 \% 10 = 7$	
7	$7 // 1 = 7$	$7 \% 1 = 0$	

Note que el algoritmo descompone el número desde el dígito más significativo al dígito menos significativo: **4 5 8 9 7**
Como determinamos la Potencia para obtener el primer dígito (P): Usando el algoritmo 1, podemos determinar la cantidad de dígitos que tiene N.

```
while N != 0:
    Cont = Cont + 1
    N = N // 10
```

Donde la variable Cont cuenta los dígitos que hay en el numero N, luego calculamos P
P = 10 ^ (Cont - 1)

Supongamos que queremos hacer una aplicación de consola que cree un numero entero formado por los dígitos pares de un numero de cedula tomándolos desde más significativo al menos significativo.

```
# Inicializamos variables
cedula = 0 # variable guarda nro cedula
dig = 0 # Variable guarda un digito
temp = 0 # Variable temporal
por = 0 # Variable que divide para obtener un digito
armaPares = 0 # Variable donde se arma un entero con los pares de cedula
contDigitos = 0 # Almacena la cantidad de digitos que hay en la cedula
# Entrada dato
cedula = int(input("De un numero de cedula: "))
# Determinamos la cantidad de digitos que hay en la cedula
temp = cedula
while temp != 0:
    temp //= 10
    contDigitos += 1
# Determinamos el valor de pot
pot = 10**(contDigitos - 1)
# Ciclo obtiene los digitos pares y crea un entero
# con los mismos desde el mas significativo al menos significativo
temp = cedula
while temp != 0:
    dig = temp // pot # Obtenemos el digito mas significativo
    # Validamos si es par
    if dig % 2 == 0:
        armaPares = armaPares*10 + dig # Armamos el par en la variable armaPares
    temp %= pot # Eliminamos el digito mas significativo obtenido
    pot //= 10 # Disminuimos en 10 pot
print("")
print("****Salida****")
print("Para cedula: ", cedula, " el armado con los pares es: ", armaPares)
print("\nFin programa")
```

Si al ejecutar damos el siguiente numero de cedula: 17366821 la salida será:

Para la cedula 17366821 el armado con los pares es: 6682