

EVALUACION

Formato ISO8583

Implementar un convertidor de mensajes al formato ISO 8583 siguiendo las siguientes especificaciones. El formato ISO 8583 se conforma por dos secciones, una cabecera que indica que campos están presentes en el mensaje y una trama de datos.

La cabecera es un campo de 8 bytes (16 nibbles) en formato big endian en el cual cada bit indica que campos están presentes en el mensaje.

Por ejemplo la cabecera FFFFFFFFFFFFFFFF indica que los 64 campos disponibles están presentes en el mensaje, mientras que la 0000000000000000 indica que no hay ningún campo presente. Como ejemplo adicional la cabecera 6100000000000000 indica que los campos 2, 3 y 8 están presentes.

La forma de leer la cabecera es convertir a binario en formato big endian y leer los bits de izquierda a derecha iniciando con el primero como 1 y el último como 64.

De esta forma 6100000000000000 se transcribe como:

01100001 00000000 00000000 00000000 00000000 00000000 00000000 00000000

Leído de izquierda a derecha están presentes los campos 2, 3 y 8

La trama de datos contempla los siguientes tipos de datos:

A: Alfanumérico de ancho fijo por ejemplo "HOLA 10 " es un mensaje de 10 caracteres alfabéticos (considera las letras, espacios y números y se padea con espacios)

N: Numérico de ancho fijo por ejemplo "0000050000" es un mensaje de 10 dígitos, representa el número 50000 y se completa con ceros a la izquierda

B: Binario, solo permite caracteres hexadecimales (0-9, A-F)

LVAR: Alfanumérico de ancho variable (de 1 a 9 caracteres) por ejemplo "4HOLA" es un mensaje de 4 caracteres que dice HOLA

LLVAR: Alfanumérico de ancho variable (de 1 a 99 caracteres) "10HOLA AMIGO" es un mensaje de 10 caracteres que dice HOLA AMIGO

LLLVAR: Alfanumérico de ancho variable (de 1 a 999 caracteres) "038ESTE ES UN MENSAJE LARGO PERO NO TANTO" es un mensaje de 38 caracteres que dice ESTE ES UN MENSAJE LARGO PERO NO TANTO

El tipo de dato que ocupa cada posición se define de forma externa al mensaje, para este caso usaremos las siguientes posiciones como válidas con sus respectivos formatos

CAMPO	Tipo de Dato
2	n ..19
3	n 6
4	n 12
5	n 12
6	n 12
7	n 10
8	n 8
9	n 8
10	n 8
11	n 6
12	n 6
13	n 4
14	n 4
15	n 4
16	n 4
17	n 4
18	n 4
19	n 3
20	n 3
21	n 3
22	n 3
23	n 3
24	n 3
25	n 2
26	n 2
27	n 1
28	n 8
29	n 8
30	n 8
31	n 8
32	n ..11
33	n ..11
34	n ..28
35	a ..37
36	a ..104

CAMPO	Tipo de Dato
37	a 12
38	a 6
39	a 2
40	a3
41	a 8
42	a 15
43	a 40
44	a ..25
45	a ..76
46	a ..999
47	a ..999
48	a ..999
49	a 3
50	a 3
51	a 3
52	b 16
53	n 18
54	a ..120
55	a ..999
56	a ..999
57	a ..999
58	a ..999
59	a ..999
60	a ..7
61	a ..999
62	a ..999
63	a ..999
64	b 16

Como leer el tipo de dato:

Ejemplos:

a ..999 significa alfanúmerico de longitud variable hasta 999 -> tipo LLLVAR formato Alfanumerico

a 12 significa alfanumérico de longitud 12 exacta

Se pide construir un programa que pueda recibir el siguiente JSON

```
{  
    "campo": "valor",  
    ...  
}
```

Y que genere la trama de datos.

Por ejemplo para el mensaje

```
{  
    "2": "12346578901234567",  
    "28": "20201005"  
}
```

El mensaje generado seria

4000001000000000171234657890123456720201005