Card Transaction Authorization

Write a program to authorize credit card transactions using a simplified version of interchange message as input and output. Note that you do not need knowledge of standard ISO 8583 to complete this test.

The input message is made up of the following parts:

1. A Message Type Indicator
2. A bitmap that indicates the presence of fields
3. Data elements, the actual data fields

Message Type Indicator

Message Type indicator is a 4-digit numeric field. Possible message types are

0100 - Authorization request message
0110 - Authorization response message

Bitmap

This field is a 2-character lower case hex encoding of a 1-byte bitmap value, indicating which data elements are present. Example:

`c8` represents `11001000` which means that, data elements, bit 1, bit 2 and bit 5 are provided in this message.

Data Elements

Data elements are actual information fields. Data elements have the following types:

NUMERIC(n)
Number field with fixed length n. This type can only contain 0-9 characters and must be left padded to n characters with zero(s).

For example, NUMERIC(6) field with a value of 10, `000010` will be populated into this field.

ALPHA(n)

Alpha numeric field with fixed length n. This type can only contain 0-9a-zA-Z and spaces and must be right padded to n characters with space(s).

For example, ALPHA(6) field with a value of CITY, `"CITY "` will be populated into this field.

LLVAR

ALPHA field with variable length. This field has two separates section, the first two characters are NUMERIC(2) indicating length of actual following ALPHA value.

For example, LLVAR has value `04CITY`. Where "CITY" is the value and "04" is its length, 4.

Here are the data elements and their type definition

Bit	Type	Data Element	Mandatory
1	LLVAR	PAN (credit card number) - 14-19 digit PAN	Yes
2	NUMERIC(4)	Expiration Date (MMYY)	Yes
3	NUMERIC(10)	Transaction amount in cents (e.g. for $10, value is 0000001000)	Yes
4	ALPHA(2)	Response code	Yes (response)
5	LLVAR	Cardholder name	
6	NUMERIC(5)	ZIP code	
7		Not used	
8		Not used	

Processing Rules

When your application receiving an authorization request, it will determine whether it will authorize this transaction. If it authorizes the transaction, it will response with the same message with Response Code “OK”. If it does not authorize (decline) the transaction, it will response with the same message with Response Code “DE”. If mandatory fields are missing from the message, it will response with the same message with Response Code “ER”.

In order for your application to response with Authorize, a message must meet the following rules:

- When Zip code is provided, a transaction is approved if amount is less than $200
- When Zip code is not provided, a transaction is approved if amount is less than $100
- Expiration Date is greater than the current date

Input and Output

Your application should allow running using command line and accept an input file name as argument or allow piping file content. The input file will contain messages with a message per line (a line ends with `\n`). For example:

```
0100e0164111111111111111112250000001000
0100e0164012888888881881122500000011000
0100ec165105105105105100122500000110001 1MASTER YODA90089
0100e0164111111111111111112180000001000
01006012250000001000
```

Let break down the first message

```
Type Bitmap PAN                Exp. Amount
0100 e0     164111111111111111 1225 0000001000
```

Application will print out the authorization response messages, with one message per line:

```
0110f0164111111111111111112250000010000K
0110f0164012888888881881122500000110000E
0110fc165105105105105100122500000110000K11MASTER YODA90089
0110f0164111111111111111112180000001000E
01107012250000001000ER
```

The following can be assumed of the input:

- Input will always be in a correct format for parsing with a correct type.
- Transaction amount is a positive number
- PAN is always between 14-19 digits

You can implement this test in any language. Please provide detail instruction on how to build, test and run your application in a readme file. Please also include any decision or any assumption you made as well. Make sure that your code is readable and well tested. You may be asked to extend this during the next interview session.

Please don’t hesitate to contact our recruiter for any question.