Mohamed Khider University – Biskra.
Faculty of Exact Sciences
Computer Science department

3rd year License
Compilers
2025/2026

Work No. 2

Exercise 01:

1) How to validate Visa Card number using Regular Expression

The valid Visa Card number must satisfy the following conditions:

1. It should be 13 or 16 digits long, new cards have 16 digits and old cards have 13 digits.
2. It should start with 4.
3. If the cards have 13 digits the next twelve digits should be any number between 0-9.
4. If the cards have 16 digits the next fifteen digits should be any number between 0-9.
5. It should not contain any alphabet or special characters.

2) How to validate MasterCard number using Regular Expression?

The valid Master Card number must satisfy the following conditions.

1. It should be 16 digits long.
2. It should start with either two digits numbers may range from 51 to 55 or four digits numbers may range from 2221 to 2720.
3. In the case of 51 to 55, the next fourteen digits should be any number between 0-9.
4. In the case of 2221 to 2720, the next twelve digits should be any number between 0-9.
5. It should not contain any alphabet or special characters

Exercise 02:

1. Provide a regular expression that validates a simplified form of email addresses.

Simplified definition:
- An email address consists of one or more fields followed by an @ symbol followed by one or more fields.
- A field consists of one or more characters (letter, number, -, _).

2. Remote file identifiers are, in most cases, as follows: *user@hostname:filename*

The parts of the identifiers consist of words, which are sequences of one or more letters and numbers. The *user* part contains a single word. The *hostname:* part consists of one or more words separated by dots (as in www.uio.no). *filename* consists of one or more words separated by slashes (/) with an optional slash at the beginning and/or end. The *'user* @' part is optional and may be omitted. The entire *'user @ hostname:*' may also be omitted, including the trailing colon ':'. The 'user @' part may not appear, but the *hostname:* part must always be present.

Exercise 03: Consider the following Lex code:

```
%%
a*b { printf( "1"): }
ca { printf( "2"); }
a*ca* { printf("3"); }
%%
```

For the input string "**abcaacacaaabbaaabcaaca**", give the output of this lexical analyzer ?

Exercise 04: Consider the following Lex code:

```
%%
aa { printf("1"); }
b?a+b? { printf("2"); }
b?a*b? { printf("3"); }
.|\n { printf("4"); }
```

1. Give an example of an input to this file that will produce **123** as output, or explain why it does not exist.
2. Give an example of an input to this file that will produce 321 as output, or explain why it does not exist

Exercise 05:

Write a lex programme that replaces each letter in a text with the next letter by reversing the case (a with B, B with c, Z with a).

Example:

Input: I feel REALLY good!

➔Output: kF NF TFOT wsbjnfou CIFO!