n° 2.1

scan (self-loop): x , x , R
scan → qa: B , B , S
scan → seek$_0$: 0 , x , L
scan → seek$_1$: 1 , x , L

seek$_0$ (self-loop): 0 , 0 , R / 1 , 1 , R
seek$_0$ → end$_0$: x , x , L / B , B , L

seek$_1$ (self-loop): 0 , 0 , R / 1 , 1 , R
seek$_1$ → end$_1$: x , x , L / B , B , L

end$_0$ → back: 0 , x , L
end$_1$ → back: 1 , x , L

back (self-loop): 0 , 0 , L / 1 , 1 , L / x , x , L
back → scan: B , B , R

n° 2.2

seek_a: x, x, R (loop)

seek_b: a, a, R / b̶, b̶, R (loop)

seek_c: b, b, R / c̶, c̶, R (loop)

back: a, a, L / b, b, L / c, c, L / a̶, a̶, L / b̶, b̶, L / c̶, c̶, L (loop)

→ seek_a

seek_a → seek_b: a, x, R

seek_b → seek_c: b, b̶, R
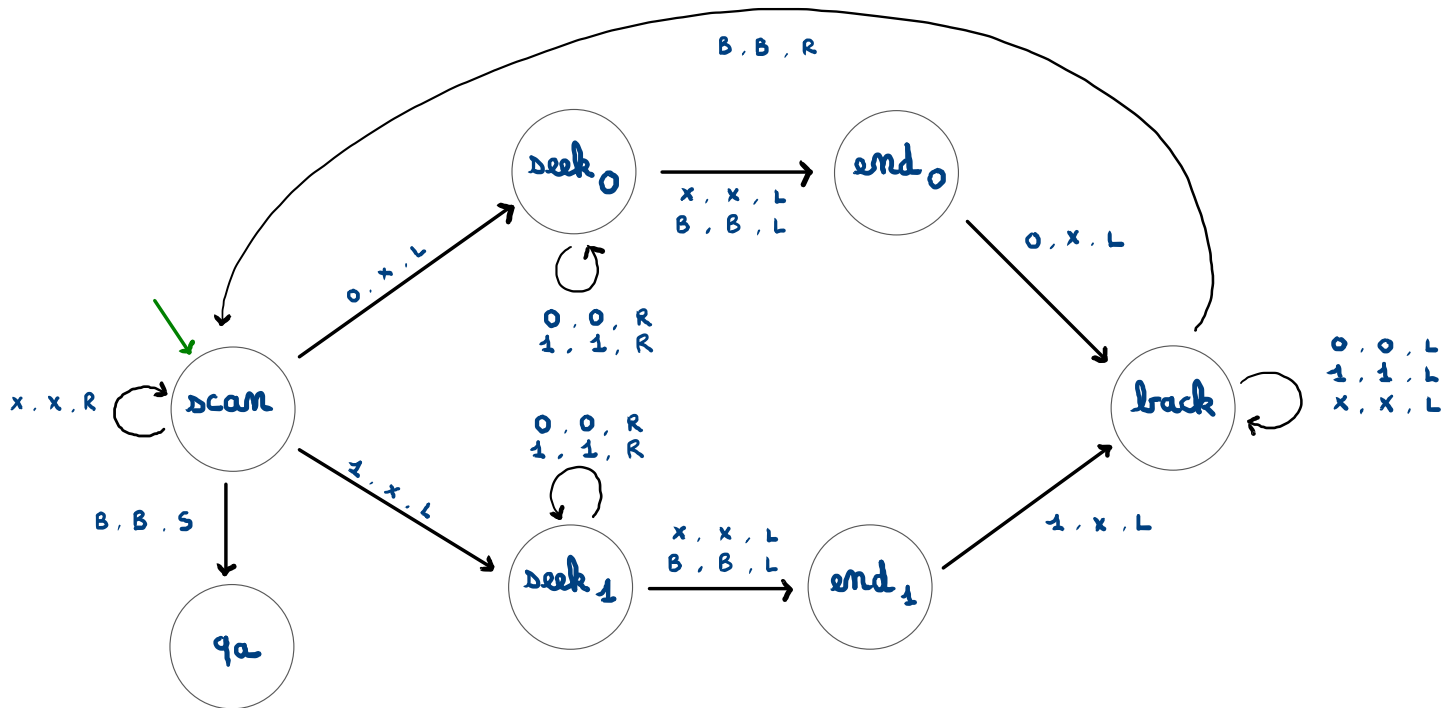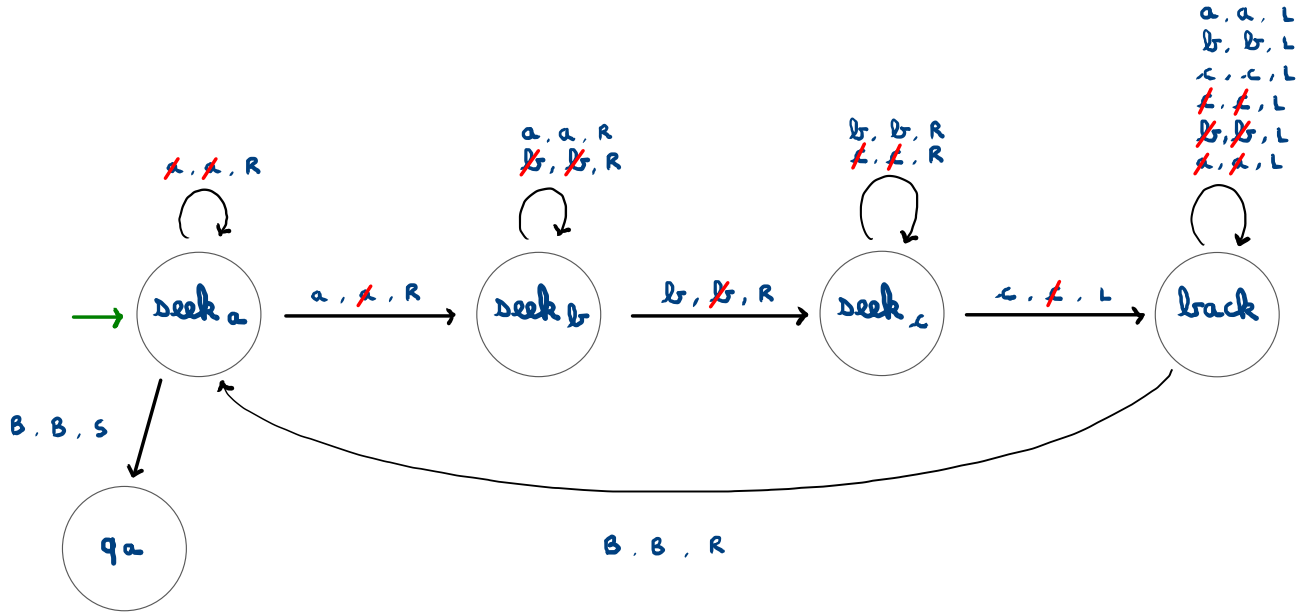
seek_c → back: c, c̶, L

seek_a → qa: B, B, S

back → seek_a: B, B, R

n° 2.3

if L' were RE, there would exist U accepting it

now consider V that on the input x

. writes 1 to the left of x

. simulates U on $1x$

V accepts $^<L$, absurd

thus L' not RE

n° 2.4

$\overline{L_j} = \bigcup_{i \neq j} L_i$ is RE   as the union of RE languages

but L_j is RE as well

thus L_j is R $\forall j$