# Regular Language to Regular Expression

Consider {a,b} as input alphabet.

1. L = {w| the length of w is exactly 2}

RE: aa + ab + ba + bb

Or RE: (a+b)(a+b)

L = {aa,ab,ba,bb}

{a,b}.{a,b}=(a+b)(a+b)

1. L = {w| the length of w is maximum 2}

RE: ɛ + a + b + aa + ab + ba + bb

Or RE: (ɛ + a + b) (ɛ + a+ b)

L = { ɛ , a, b, aa, ab, ba, bb}

(ɛ + a + b) (ɛ + a+ b)

1. L = {w| the length of w is minimum 2}

RE: (a + b) (a + b) (a+b)\*

Or RE: (a+b) (a+b)+

1. L = {w| w is a string of even length}/ |w| mod 2 = 0

((a+b)(a+b))\* ɛ

1. L = {w| w is a string of odd length}/ |w| mod 2 = 1

((a+b)(a+b))\*(a+b)

1. L = {w| w where |w| mod 3=0}

((a +b) (a+b) (a+b))\*

1. L = {w| w starts with ‘a’}

a(a+b)\*

1. L = {w| w ends with ‘a’}

(a+b)\*a

1. L = {w| w does contain ‘a’}

(a+b)\*a(a+b)\*

1. L = {w| w starts with ‘ab’}

ab(a+b)\*

1. L = {w| w ends with ‘ab’}

(a+b)\*ab

1. L = {w| w does contain ‘ab’}

(a+b)\*ab(a+b)\*

1. L = {w| w has even number of a’s}
2. L = {w| w where na(w) mod 3=1}
3. L = {w| w where na(w) = 3}
4. L = {w| w starts with a and end b}

a(a+b)\*b

1. L = {w| w starts and end with same symbol}

a(a+b)\*a + b(a+b)\*b + ɛ + a + b

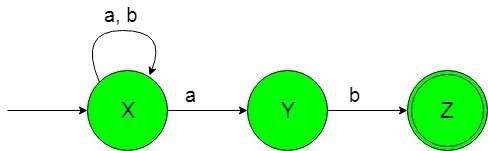
1. L = {w| w starts and end with different symbol}

a(a+b)\*b + b(a+b)\*a

1. L = {w| where every ‘a’ should followed by ‘b’}
2. L = {w| where every ‘a’ should never be followed by ‘b’}
3. L = {w| where every ‘a’ should followed by ‘bb’}
4. L = {w| where every ‘a’ should never be followed by ‘bb’}
5. L = {w| w where ‘w’ either begins or ends with ab}
6. L = {w| w does contain three consecutive b’s}

(a+b)\*bbb(a+b)\*

1. a = {a}
2. b = {b}
3. ɛ = {ɛ}
4. ∅ = {}
5. a + b = {a, b}
6. ab + ba = {ab , ba}
7. a\* = set of all possible strings over a including ɛ
8. a+ = set of all possible strings over a excluding ɛ
9. (a+b)\* =set of all possible strings over a,b including ɛ
10. (a+b)+= set of all possible strings over a,b excluding ɛ



(a+b)\*ab + ɛ

(a+b)\*(a+b)\*