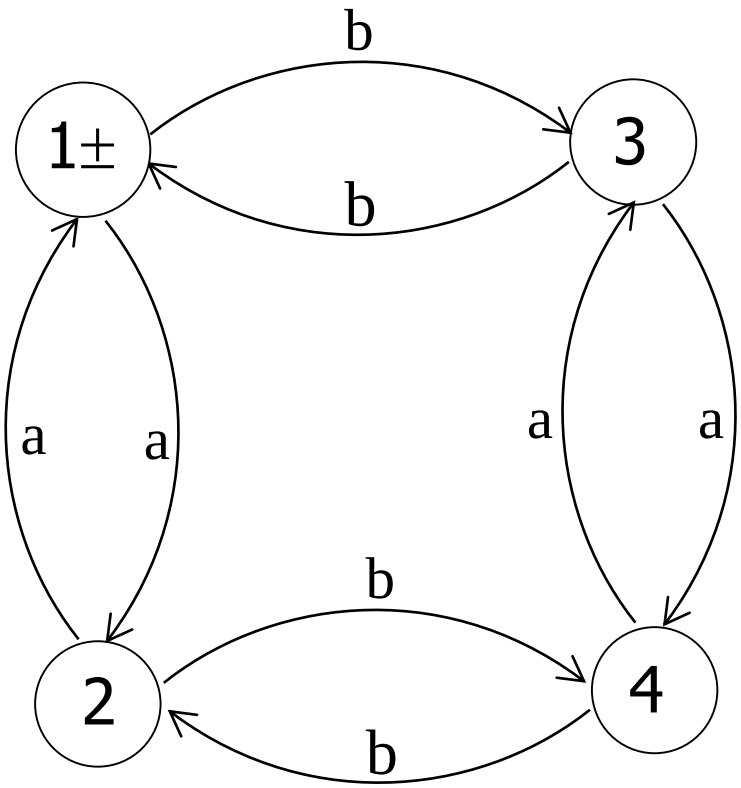
Recap lecture 6

- Language of strings, beginning with and ending in different letters, Accepting all strings, accepting non-empty strings, accepting no string, containing double a's, having double 0's or double 1's, containing triple a's or triple b's, EVEN-EVEN

Example EVEN-EVEN continued ...



FA corresponding to finite languages

- **Example**

  Consider the language

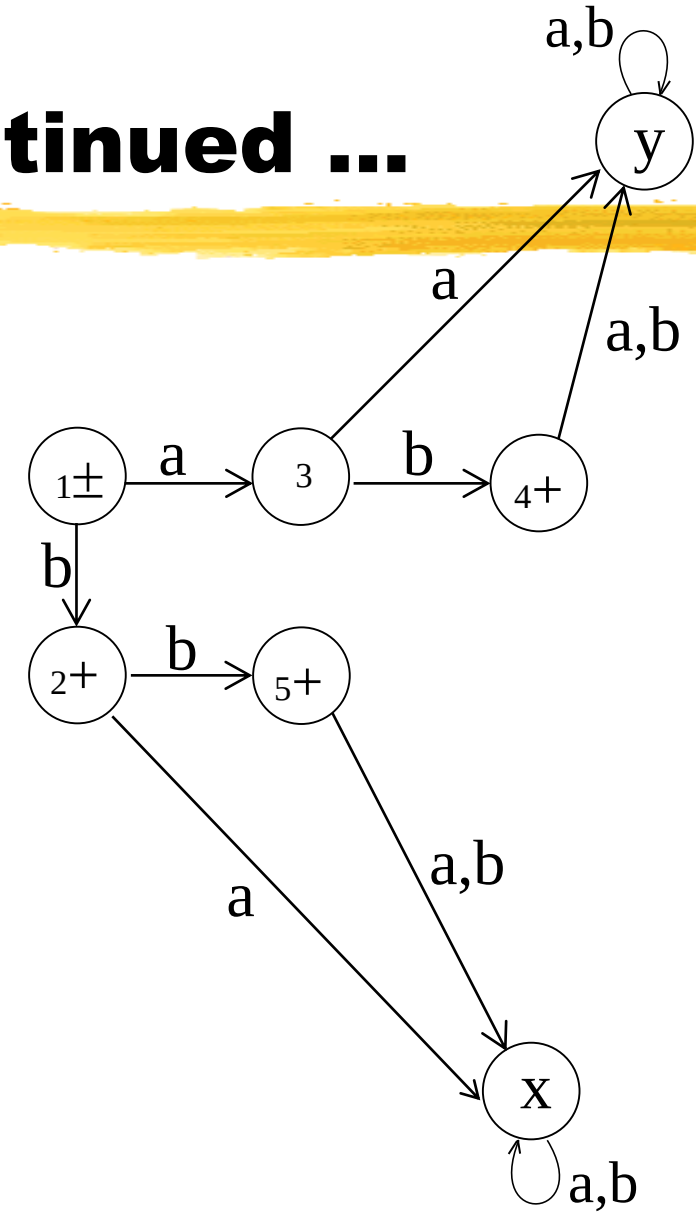
  L = {Λ, b, ab, bb}, defined over

  Σ ={a, b}, expressed by

  Λ + b + ab + bb OR Λ + b (Λ + a + b).

  The language L may be accepted by the following FA

Example continued ...

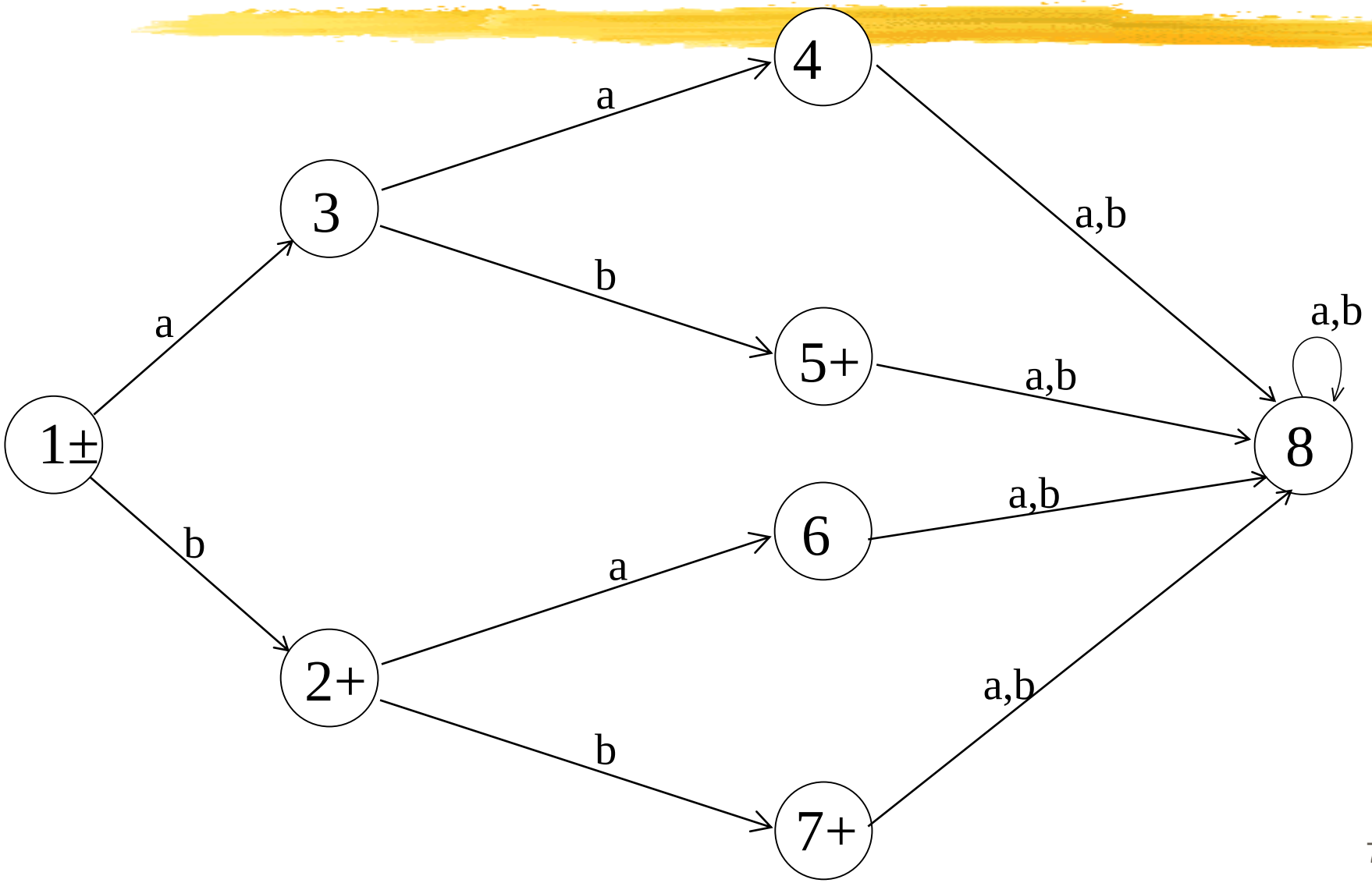


Example Continued …

- It is to be noted that the states x and y are called **Dead States, Waste Baskets or Davey John Lockers,** as the moment one enters these states there is no way to leave it.

Note

- It is to be noted that to build an FA accepting the language having less number of strings, the tree structure may also help in this regard, which can be observed in the following transition diagram for the Language L, discussed in the previous example


4
a
3
a,b
b
a,b
a
5+
a,b
1±
8
a,b
b
6
a
2+
a,b
b
7+

Example

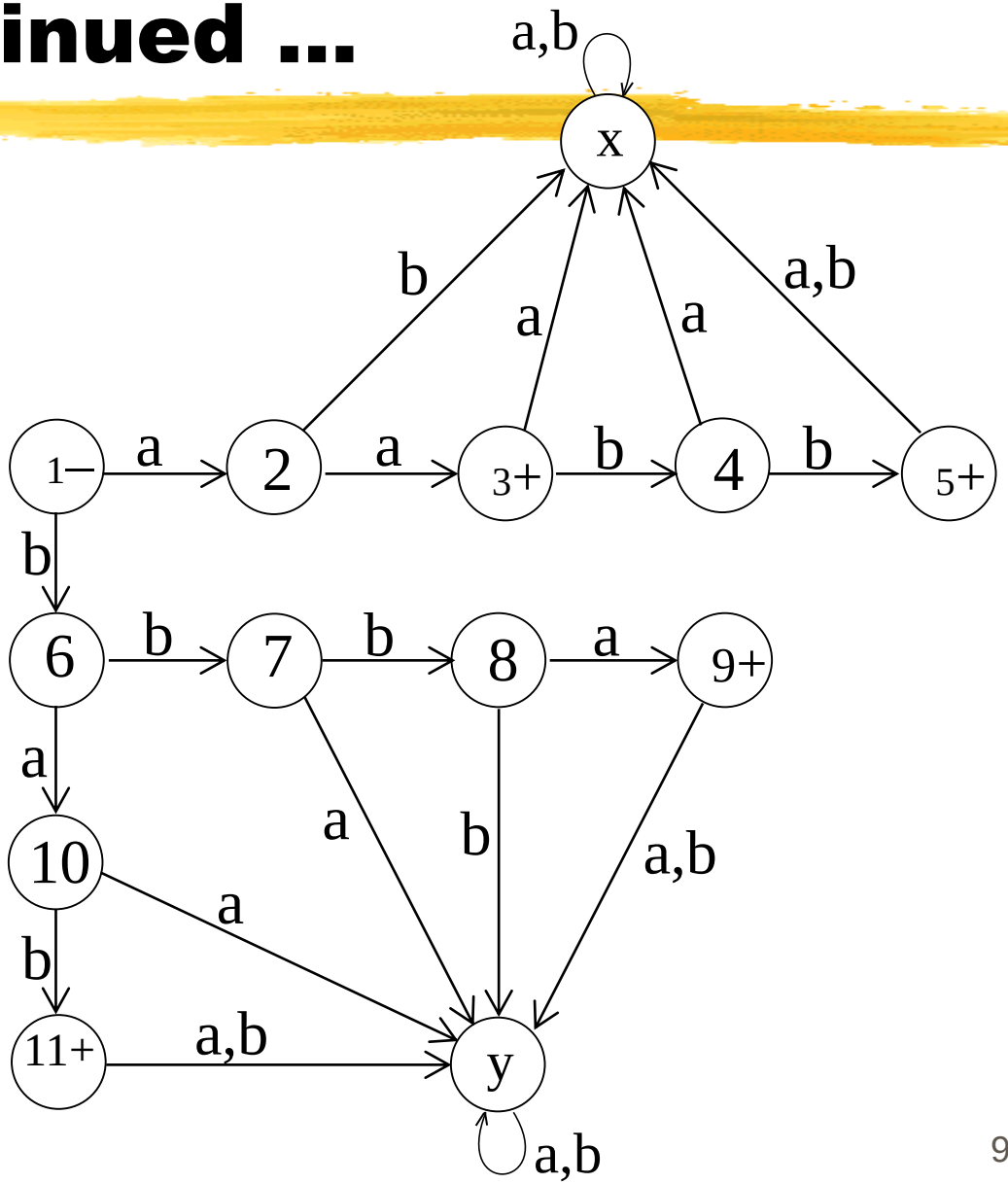
Consider the language

L = {aa, bab, aabb, bbba}, defined over

Σ ={a, b}, expressed by
aa + bab + aabb + bbba
OR aa (Λ + bb) + b (ab + bba)

The above language may be accepted by the following FA

Example Continued ...



Example

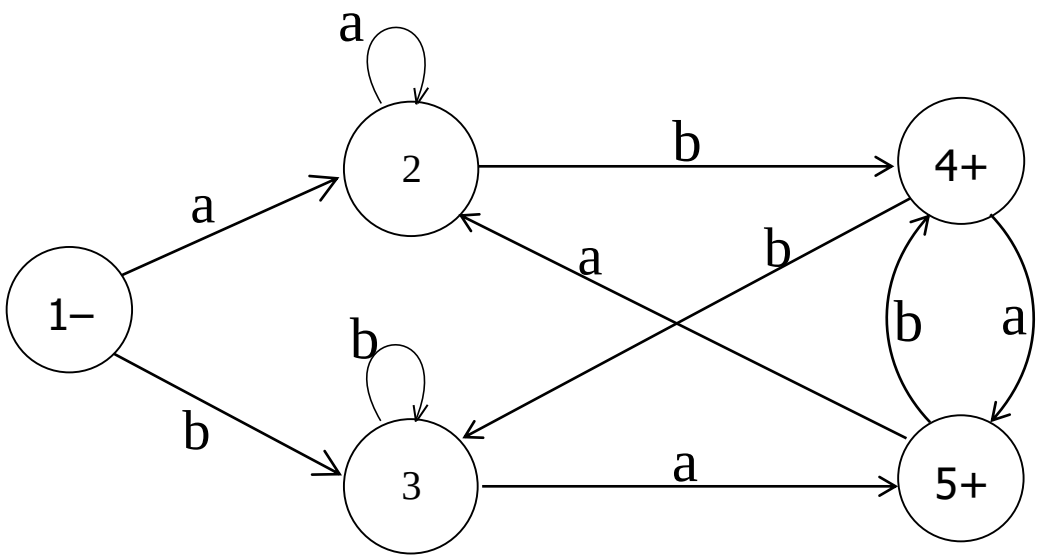
Consider the language
L = {w belongs to $\{a,b\}^*$: length(w) >= 2 and w neither ends in **aa** nor **bb**}.

The language L may be expressed by the regular expression

$$(a+b)^*(ab+ba)$$

This language may be accepted by the following FA

Example Continued ...



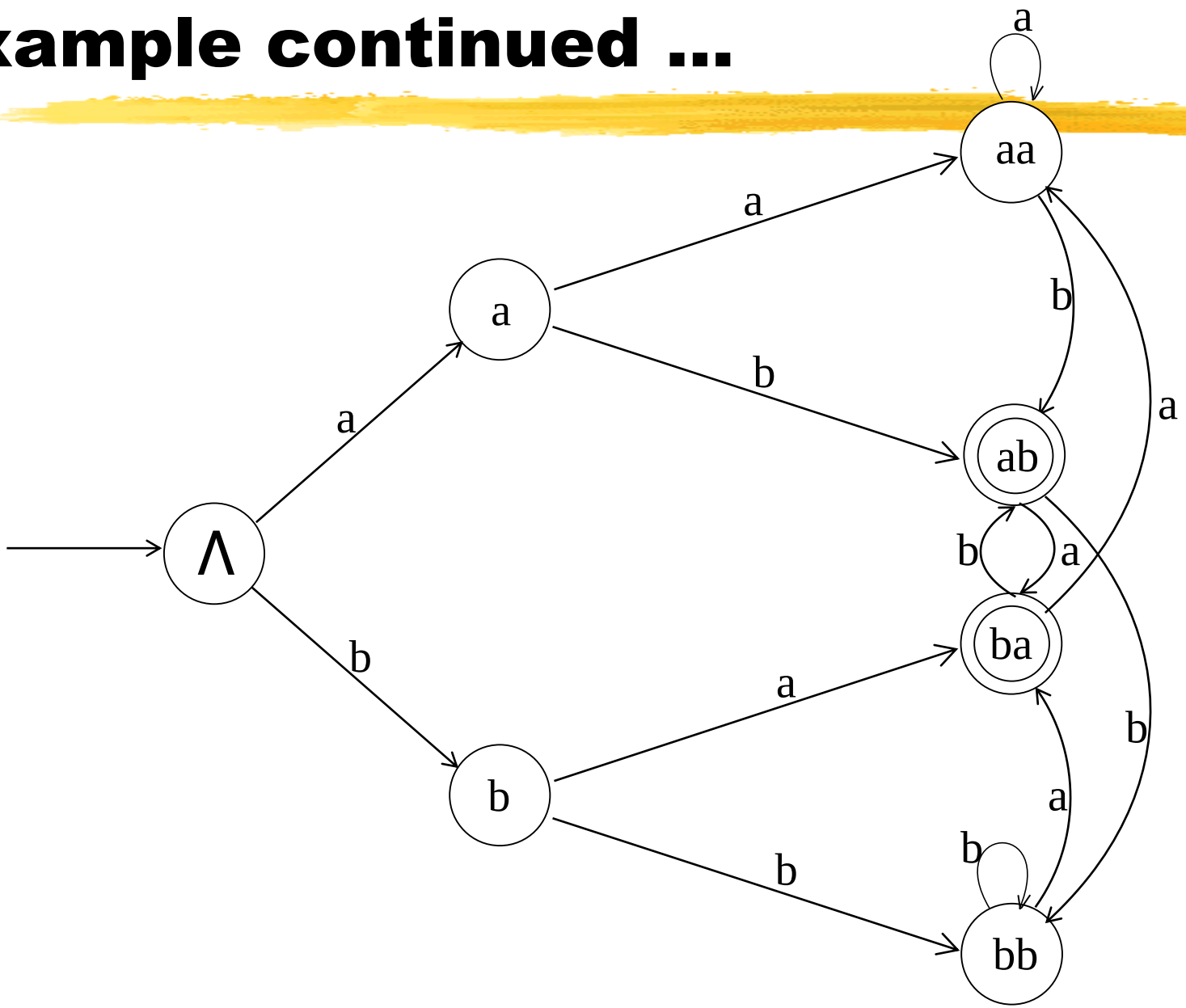
Note

- It is to be noted that building an FA corresponding to the language L, discussed in the previous example, seems to be quite difficult, but the same can be done using tree structure along with the technique discussed in the book

  Introduction to Languages and Theory of Computation, by J. C. Martin

  so that the strings ending in aa, ab, ba and bb should end in the states labeled as aa, ab, ba and bb, respectively;as shown in the following FA

Example continued ...



Example

- Consider the language

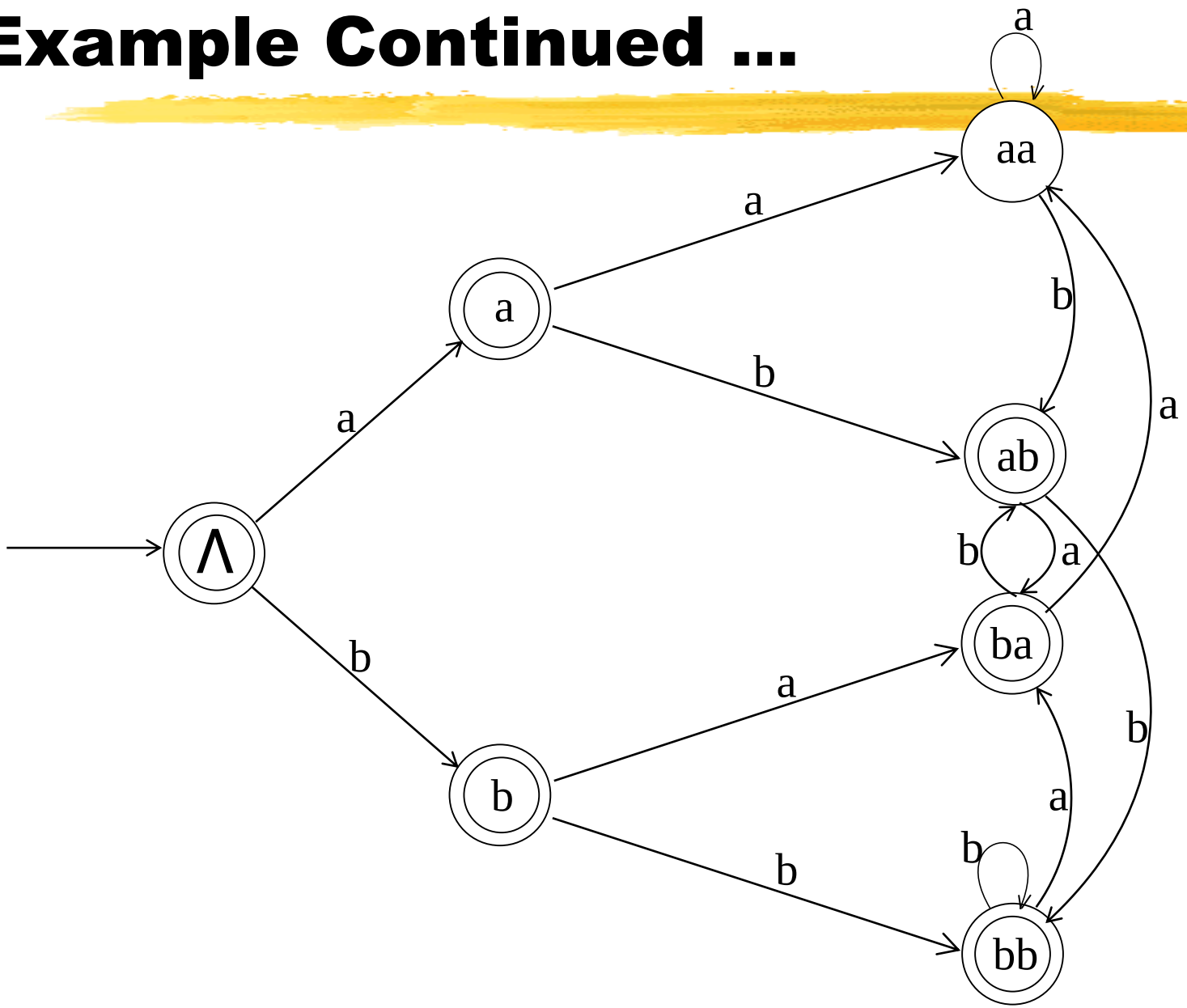
  L = {w belongs to {a,b}*: w does not end in **aa**}.

  The language L may be expressed by the regular expression

  $$\Lambda + a + b + (a+b)^*(ab+ba+bb)$$

  This language may be accepted by the following FA

Example Continued ...



Task

- Using the technique discussed by Martin, build an FA accepting the following language

  L = {w belongs to $\{a,b\}^*$: length(w) >= 2 and second letter of w, from right is a}.

Task

- Using the technique discussed by Martin, build an FA accepting the following language

  L = {w belongs to {a,b}: w neither ends in **ab** nor **ba**}.

Defining Languages (continued)…

- **Method 5 (Transition Graph)**

  Definition: A Transition graph (TG), is a collection of the followings

  1) Finite number of states, at least one of which is start state and some (maybe none) final states.
  2) Finite set of input letters (Σ) from which input strings are formed.
  3) Finite set of transitions that show how to go from one state to another based on reading specified substrings of input letters, possibly even the null string Λ.

Defining Languages (continued)…

- **Method 5 (Transition Graph)**

  Definition: A Transition graph (TG), is a collection of the followings

  1) Finite number of states, at least one of which is start state and some (maybe none) final states.
  2) Finite set of input letters (Σ) from which input strings are formed.
  3) Finite set of transitions that show how to go from one state to another based on reading specified substrings of input letters, possibly even the null string (Λ).

Summing up …

- **EVEN EVEN,** FA corresponding to finite languages(using both methods), Transition graphs.