

*Pharos University in Alexandria*  
*Faculty of Computer Science & Artificial Intelligence*  
*Course Title: Theory of Computation*  
*Code: CS 307*



# Theory of Computation

**Lecturer: Sherine Shawky**

Text Books

1. Introduction to formal languages and automata, Peter Linz, 6th edition, 2017.

Week 6

Closure Properties  
and  
Pumping Lemma

# Closure Properties

- Closure properties on regular languages are defined as certain operations on regular language that are guaranteed to produce regular language.
- Closure refers to some operation on a language, resulting in a new language that is of the same “type” as originally operated on i.e., regular.
- Regular languages are closed under the following operations:

# Closure Properties

- In an automata theory, there are different closure properties for regular languages. They are as follows :
  - Union
  - Intersection
  - concatenation
  - Kleene closure
  - Complement

# Union

- If  $L_1$  and  $L_2$  are two regular languages, their union  $L_1 \cup L_2$  will also be regular.

## Example

- $L_1 = \{a^n \mid n > 0\}$  and  $L_2 = \{b^n \mid n > 0\}$
- $L_3 = L_1 \cup L_2 = \{a^n \cup b^n \mid n > 0\}$  is also regular.

# Intersection

- If  $L_1$  and  $L_2$  are two regular languages, their intersection  $L_1 \cap L_2$  will also be regular.

## Example

- $L_1 = \{a^m b^n \mid n > 0 \text{ and } m > 0\}$  and
- $L_2 = \{a^m b^n \cup b^n a^m \mid n > 0 \text{ and } m > 0\}$
- $L_3 = L_1 \cap L_2 = \{a^m b^n \mid n > 0 \text{ and } m > 0\}$  are also regular.

# Concatenation

- If  $L_1$  and  $L_2$  are two regular languages, their concatenation  $L_1.L_2$  will also be regular.

## Example

- $L_1 = \{a^n \mid n > 0\}$  and  $L_2 = \{b^n \mid n > 0\}$
- $L_3 = L_1.L_2 = \{a^m . b^n \mid m > 0 \text{ and } n > 0\}$  is also regular.

# Kleene Closure

- If  $L_1$  is a regular language, its Kleene closure  $L_1^*$  will also be regular.

## Example

- $L_1 = (a \cup b)$
- $L_1^* = (a \cup b)^*$

# Complement

- If  $L(G)$  is a regular language, its complement  $L'(G)$  will also be regular. Complement of a language can be found by subtracting strings which are in  $L(G)$  from all possible strings.

## Example

- $L(G) = \{an \mid n > 3\}$   $L'(G) = \{an \mid n \leq 3\}$

# Pumping Lemma for RL

- The language accepted by Finite Automata is known as Regular Language.
- Pumping Lemma is used to prove that a Language is not Regular.
- It cannot be used to prove that a language is Regular.

# What is Pumping Lemma?

- The term Pumping Lemma is made up of two words:-
  - Pumping: The word pumping refers to generating many input strings by pushing a symbol in an input string repeatedly.
  - Lemma: The word Lemma refers to the intermediate theorem in a proof.

# Types of Pumping Lemma

- There are two Pumping Lemmas, that are defined for
  - Regular Languages
  - Context-Free Languages

# Pumping Lemma for RL

- **Theorem:**

- If A is a Regular Language, then A has a Pumping Length 'P' such that any string 'S' where  $|S| \geq P$  may be divided into three parts  $S = xyz$  such that the following conditions must be true:
  - **1.)  $xy^iz \in A$  for every  $i \geq 0$**
  - **2.)  $|y| > 0$**
  - **3.)  $|xy| \leq P$**
- In simple words, if a string y is 'pumped' or insert any number of times, the resultant string still remains in A.

# Pumping Lemma for RL

- Pumping Lemma is used as proof of the **irregularity** of a language.
- It means, that if a language is regular, it always satisfies the pumping lemma.
- If at least one string is made from pumping, not in language A, then A is not regular.

# Pumping Lemma for RL

- We use the **CONTRADICTION** method to prove that a language is not Regular.
- **Steps to prove that a language is not Regular using Pumping Lemma:**
  - **Step 1:** Assume that Language A is Regular.
  - **Step 2:** It has to have a Pumping Length (say P).
  - **Step 3:** All strings longer than P can be pumped  $|S| \geq P$ .
  - **Step 4:** Now, find a string 'S' in A such that  $|S| \geq P$ .

# Pumping Lemma for RL

- **Step 5:** Divide  $S$  into  $x y z$  strings.
  - **Step 6:** Show that  $xy^iz \notin A$  for some  $i$ .
  - **Step 7:** Then consider how  $S$  can be divided into  $x y z$ .
  - **Step 8:** Show that none of the above strings satisfies all three pumping conditions simultaneously.
  - **Step 9:**  $S$  cannot be pumped  $\implies$  CONTRADICTION.
- 
- Let's apply these above steps to check whether a Language is not Regular with the help of Pumping Lemma.

# Implementation of Pumping lemma for RL

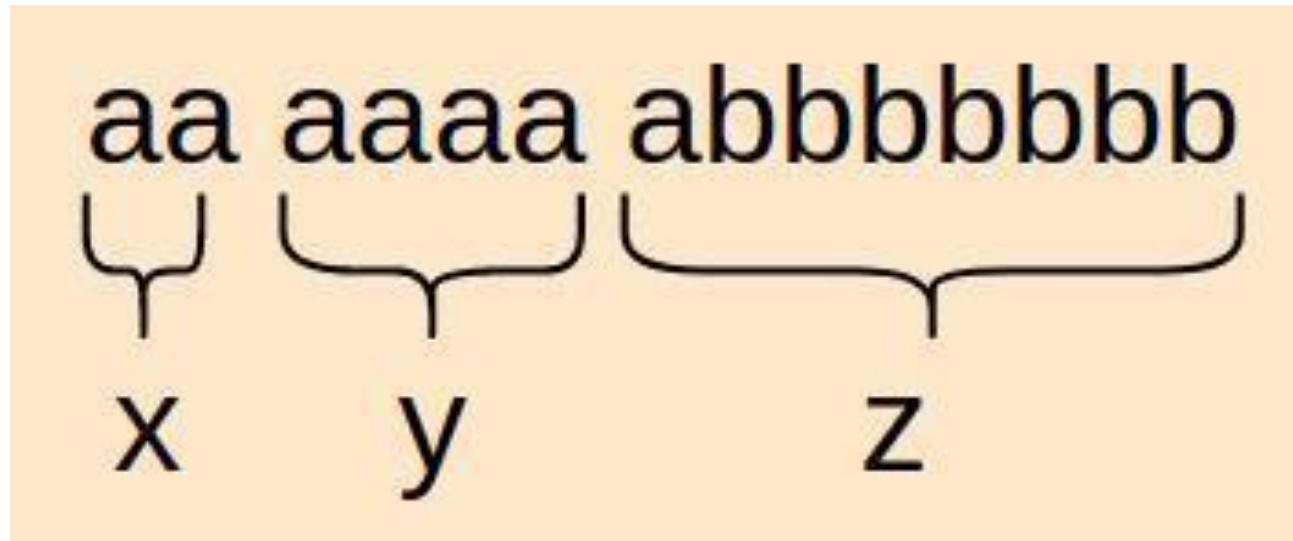
- **Example:**
- Using Pumping Lemma, prove that the language  $A = \{a^n b^n \mid n \geq 0\}$  is **Not Regular**.

# Implementation of Pumping lemma for RL

- **Example:**
- Using Pumping Lemma, prove that the language  $A = \{a^n b^n \mid n \geq 0\}$  is **Not Regular**.
- **Solution:** We will follow the steps we have learned above to prove this.
- Assume that A is Regular and has a Pumping length = P.
- Let a string  $S = a^p b^p$ .
- Now divide the S into the parts, x y z.
- To divide the S, let's take the value of  $P = 7$ .
- Therefore,  $S = aaaaaaabb bbbbbb$  (by putting  $P=7$  in  $S = a^p b^p$ ).

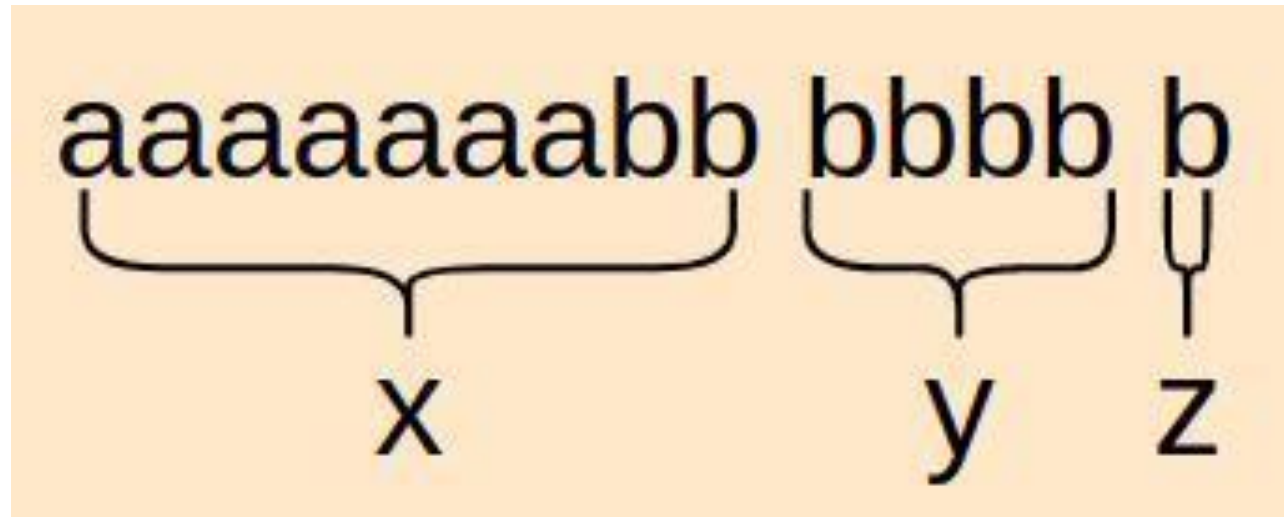
# Pumping Lemma for RL

- Case 1: Y consists of a string having the letter only 'a'.



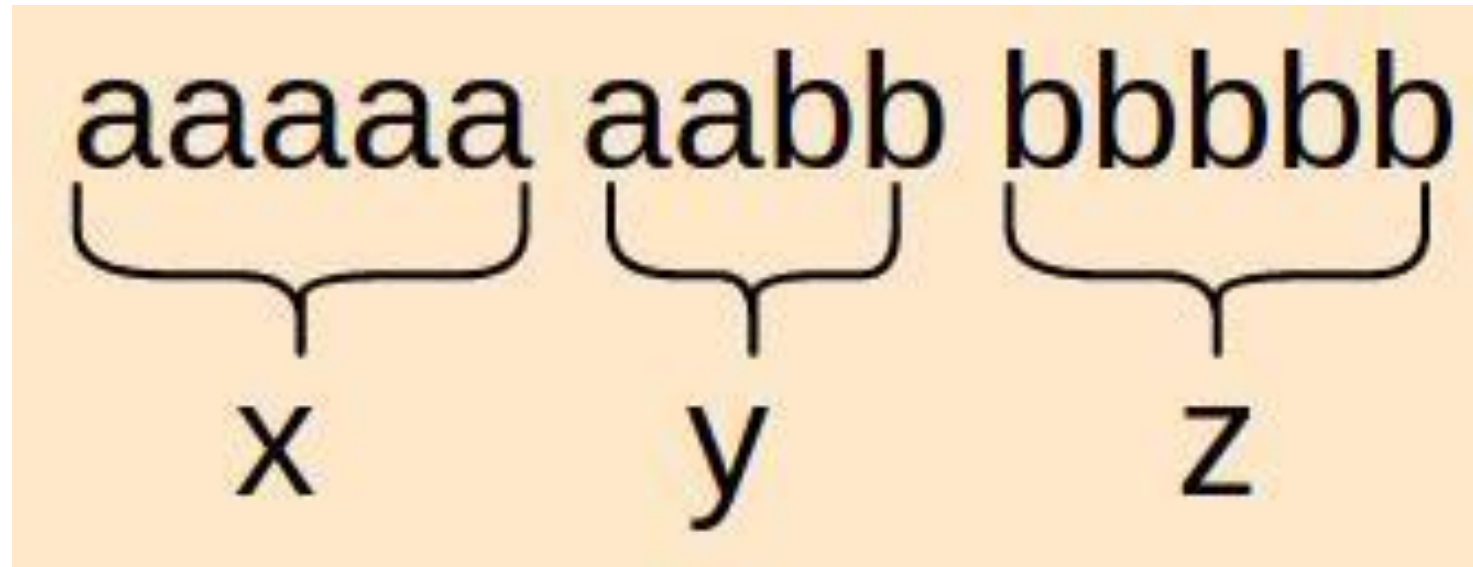
# Pumping Lemma for RL

- Case 2: Y consists of a string having the letter only 'b'.



# Pumping Lemma for RL

- Case 3: Y consists of a string with the letters 'a' and 'b'.



# Pumping Lemma for RL

- For all the above cases, we need to show  $xy^iz \notin A$  for some  $i$ .
- Let the value of  $i = 2$ .  $xy^iz \Rightarrow xy^2z$
- In Case 1.  $xy^2z = \mathbf{aa\ aaaa\ aaaa\ abbbbbbb}$
- No of 'a' = 11, No. of 'b' = 7.
- Since the **No of 'a'  $\neq$  No. of 'b'**, but the original language has an equal number of 'a' and 'b'; therefore, this string will not lie in our language.

# Pumping Lemma for RL

- In Case 2.  **$xy^2z = \text{aaaaaaabb bbbb bbbb b}$**
- No of 'a' = 7, No. of 'b' = 11.
- Since the **No of 'a'  $\neq$  No. of 'b'**, but the original language has an equal number of 'a' and 'b'; therefore, this string will not lie in our language.

# Pumping Lemma for RL

- In Case 3.  **$xy^2z = \text{aaaa aabb aabb bbbbbb}$**
- No of 'a' = 8, No. of 'b' = 9.
- Since the **No of 'a'  $\neq$  No. of 'b'**, but the original language has an equal number of 'a' and 'b', and also, this string did not follow the  $a^n b^n$  pattern; therefore, this string will not lie in our language.
- We can see at  $i = 2$  all the above three strings do not lie in the language  $A = \{a^n b^n \mid n \geq 0\}$ .
- Therefore, the language  $A = \{a^n b^n \mid n \geq 0\}$  is not Regular.