CHAPTER 2
FINITE AUTOMATA

4. Languages that are Not Regular
5. State Minimization
CHAPTER 2
PART 4: Languages that are not Regular
Languages that are Not Regular

We know that there are uncountably many and exactly $C$ of all languages over any alphabet $\Sigma \neq \emptyset$.

We also know that there are only $\aleph_0$, i.e. infinitely countably many regular languages.

It means that we have uncountably many and exactly $C$ languages that are not regular.

Reminder

A language $L \subseteq \Sigma^*$ is regular if and only if there is a regular expression $r \in R$ that represents $L$, i.e. such that

$$L = \mathcal{L}(r)$$
Regular or not Regular Languages

We look now at some simple examples of languages that might be, or not be regular

E1 The language \( L_1 = a^*b^* \) is regular because it is defined by a regular expression

E2 The language

\[ L_2 = \{a^n b^n : n \geq 0\} \subseteq L_1 \]

is not regular

We will prove it using a very important theorem to be proved that is called **Pumping Lemma**
Regular or not Regular Languages

Intuitively we can see that

\[ L_2 = \{a^n b^n : n \geq 0\} \]

can’t be regular as we can’t construct a finite automaton accepting it.
Such automaton would need to have something like a memory to store, count and compare the number of a’s with the number of b’s.
Regular or not Regular Languages

We will define and study in Chapter 3 a new class of automata that would accommodate the "memory" problem. They are called **Push Down Automata**.

We will prove that they accept a larger class of languages, called **context free** languages.
Regular or not Regular Languages

**E3** The language \( L_3 = a^* \) is regular because is defined by a regular expression.

**E4** The language \( L_4 = \{a^n : n \geq 0\} \) is regular because in fact \( L_3 = L_4 \).

**E5** The language \( L_4 = \{a^n : n \in \text{Prime}\} \) is not regular. We will prove it using Pumping Lemma.
Regular or not Regular Languages

E6  The language \( L_6 = \{ a^n : n \in \text{EVEN} \} \) is regular because in fact \( L_6 = (aa)^* \)

E7  The language \( L_7 = \{ w \in \{a,b\}^* : w \text{ has an equal number of } a' \text{'s and } b' \text{'s } \} \) is not regular

Proof
Assume that \( L_7 \) is regular
We know that \( L_1 = a^*b^* \) is regular
Hence the language \( L = L_7 \cap L_1 \) is regular, as the class of regular languages is closed under intersection
But obviously, \( L = \{ a^n b^n : n \in \mathbb{N} \} \) and was proved to be not regular
This contradiction proves that \( L_7 \) is not regular
Regular aor not Regular Languages

E8 The language \( L_8 = \{ ww^R : \ w \in \{ a, b \}^* \} \)

is **not regular**

We prove it using **Pumping Lemma**

E9 The language \( L_9 = \{ ww : \ w \in \{ a, b \}^* \} \)

is **not regular**

We prove it using **Pumping Lemma**
Regular or not Regular Languages

\[ E_{10} \quad \text{The language} \quad L_{10} = \{wcw : \ w \in \{a, b\}^*\} \]

is not regular
We prove it using Pumping Lemma

\[ E_{11} \quad \text{The language} \quad L_{11} = \{w\overline{w} : \ w \in \{a, b\}^*\} \]

where \( \overline{w} \) stands for \( w \) with each occurrence of \( a \) is replaced by \( b \), and vice versa

is not regular
We prove it using Pumping Lemma
Regular or not Regular Languages

E12 The language

\[ L_{12} = \{ xy \in \Sigma^* : x \in L \text{ and } y \notin L \text{ for any regular } L \subseteq \Sigma^* \} \]

is regular

Proof Observe that \( L_{12} = L \circ \overline{L} \) where \( \overline{L} \) denotes a complement of \( L \), i.e.

\[ \overline{L} = \{ w \in \Sigma^* : w \in \Sigma^* - L \} \]

\( L \) is regular, and so is \( \overline{L} \), and \( L_{12} = L \circ \overline{L} \) is regular by the following, already proved theorem

Closure Theorem The class of languages accepted by Finite Automata FA is closed under \( \cup, \cap, -, \circ, * \)
Regular or not Regular Languages

**E13** The language

\[ L_{13} = \{ w^R : \ w \in L \ \text{and} \ L \ \text{is regular} \} \]

is **regular**

**Definition** For any language \( L \) we call the language

\[ L_R = \{ w^R : \ w \in L \} \]

the **reverse** language of \( L \)

The **E13** says that the following holds

**Fact**

For any **regular** language \( L \), its **reverse** language \( L^R \) is **regular**
Regular or not Regular Languages

Fact
For any regular language $L$, its reverse language $L^R$ is regular.

Proof Let $M = (K, \Sigma, \Delta, s, F)$ be such that $L = L(M)$.
The reverse language $L^R$ is accepted by a finite automata

$$M^R = (K \cup s', \Sigma, \Delta', s', F = \{s\})$$

where $s' \notin K$ and

$$\Delta' = \{(r, w, p) : (p, w, r) \in \Delta, \ w \in \Sigma^*\} \cup \{(s', e, q) : q \in F\}$$

We used the Lecture Definition of $M$.
Regular and NOT Regular Languages

Proof of \textbf{E13} pictures

\textbf{Diagram} of $M$

\textbf{Diagram} of $M^R$
Regular and NOT Regular Languages

E14
Any finite language is regular

Proof Let $L \subseteq \Sigma^*$ be a finite language, i.e.

$$L = \emptyset \text{ or } L = \{w_1, w_2, \ldots w_n\} \text{ for } n > 0$$

We construct the finite automata $M$ such that

$$L(M) = L = \{w_1\} \cup \{w_2\} \cup \ldots \{w_n\} = L_{w_1} \cup \ldots \cup L_{w_n}$$

as $M = M_{w_1} \cup \ldots \cup M_{w_n} \cup M_{\emptyset}$

where

![Diagram of a finite automaton with states and transitions](image)
Exercises

Exercise 1
Show that the language

\[ L = \{ xyx^R : \ x, y \in \Sigma \} \]

is regular for any \( \Sigma \)
Exercises

Exercise 1
Show that the language
\[ L = \{ xyx^R : x, y \in \Sigma \} \]
is regular for any \( \Sigma \)

Proof
For any \( x \in \Sigma, x^R = x \)
\( \Sigma \) is a finite set, hence
\[ L = \{ xyx : x, y \in \Sigma \} \]
is also finite and we just proved that any finite language is regular
Exercises

Exercise 2
Show that the class of regular languages is not closed with respect to subset relation.

Exercise 3
Given \( L_1, L_2 \) regular languages, is \( L_1 \cap L_2 \) also a regular language?
Exercises

Exercise 2
Show that the class of regular languages is not closed with respect to subset relation.

Solution
Consider two languages

\[ L_1 = \{a^n b^n : \ n \in \mathbb{N}\} \quad \text{and} \quad L_2 = a^* b^* \]

Obviously, \( L_1 \subseteq L_2 \) and \( L_1 \) is a non-regular subset of a regular \( L_2 \)

Exercise 3
Given \( L_1, L_2 \) regular languages, is \( L_1 \cap L_2 \) also a regular language?

Solution
YES, it is because the class of regular languages is closed under \( \cap \)
Exercises

Exercise 4
Given $L_1, L_2$, such that $L_1 \cap L_2$ is a regular language
Does it imply that both languages $L_1, L_2$ must be regular?
Exercises

Exercise 4
Given $L_1, L_2$, such that $L_1 \cap L_2$ is a regular language
Does it imply that both languages $L_1, L_2$ must be regular?

Solution
NO, it doesn’t. Take the following $L_1, L_2$

$L_1 = \{a^n b^n : n \in N\}$ and $L_2 = \{a^n : n \in \text{Prime}\}$

The language $L_1 \cap L_2 = \emptyset$ is a regular language none of $L_1, L_2$ is regular
Exercises

Exercise 5
Show that the language

\[ L = \{ xyx^R : \ x, y \in \Sigma^* \} \]

is regular for any \( \Sigma \).
Exercises

Exercise 5

Show that the language

\[ L = \{ xyx^R : x, y \in \Sigma^* \} \]

is regular for any \( \Sigma \)

Solution

Take a case of \( x = e \in \Sigma^* \)

We get a language

\[ L_1 = \{ eye^R : e, y \in \Sigma^* \} \subseteq L \]

and of course \( L_1 = \Sigma^* \) and so \( \Sigma^* \subseteq L \subseteq \Sigma^* \)

Hence \( L = \Sigma^* \) and \( \Sigma^* \) is regular

This proves that \( L \) is regular
Exercises

Exercise 6
Given a regular language $L \subseteq \Sigma^*$
Show that the language

$$L_1 = \{xy \in \Sigma^* : x \in L \quad \text{and} \quad y \notin L\}$$

is also regular
Exercises

Exercise 6

Given a regular language \( L \subseteq \Sigma^* \)

Show that the language

\[
L_1 = \{ xy \in \Sigma^* : \ x \in L \text{ and } y \notin L \}
\]

is also regular

Solution

Observe that \( L_1 = L \circ (\Sigma^* - L) \)

\( L \) is regular, hence \( (\Sigma^* - L) \) is regular (closure under complement), and so is \( L_1 \) by closure under concatenation.
For quiz 3

Read **Pumping Lemma** statement and information about its role - you need to know it for **Quiz 3**

The **proof** of the **Pumping Lemma** and its applications will not be on the Quiz 3

You will have to know it for **Quiz 4**
Review Questions
Review Questions

Write SHORT answers

Q1
For any language \( L \subseteq \Sigma^*, \Sigma \neq \emptyset \) there is a deterministic automata \( M \), such that \( L = L(M) \)

Q2
Any regular language has a finite representation.

Q3
Any finite language is regular

Q4
Given \( L_1, L_2 \) languages over \( \Sigma \), then
\[
((L_1 \cap (\Sigma^* - L_2)) \cup L_2)L_1
\]
is a regular regular language.
Review Questions

SHORT answers

Q1
For any language \( L \subseteq \Sigma^*, \Sigma \neq \emptyset \) there is a deterministic automata \( M \), such that \( L = L(M) \)
True only when \( L \) is regular

Q2
Any regular language has a finite representation.
True by definition of regular language and the fact that regular expression is a finite string

Q3
Any finite language is regular
True as we proved it

Q4
Given \( L_1, L_2 \) languages over \( \Sigma \), then
\[ (((L_1 \cap (\Sigma^* - L_2)) \cup L_2)L_1 \] is a regular regular language
True only when both are regular languages
Review Questions for Quiz

Write SHORT answers

Q5
For any finite automata $M$

$$L(M) = \bigcup \{R(1, j, n) : q_j \in F\}$$

Q6
$\Sigma$ in any Generalized Finite Automaton includes some regular expressions

Q7
Pumping Lemma says that we can always prove that a language is not regular

Q8
$L = \{a^n c^n : n \geq 0\}$ is regular
Review Questions

SHORT answers

Q5
For any finite automata $M$

$$L(M) = \bigcup \{ R(1,j,n) : q_j \in F \}$$

True only when $M$ has $n$ states and they are put in 1-1 sequence and $q_1 = s$

Q6
$\Sigma$ in any Generalized Finite Automaton includes some regular expressions

True by definition
Review Questions

Q7
Pumping Lemma says that we can always prove that a language is not regular
Not True PL serves as a tool for proving that some languages are not regular

Q8
$L = \{a^n c^n : n \geq 0\}$ is regular
Not True we proved by PL that it is not regular
PUMPING LEMMA
Pumping Lemma

**Pumping Lemma** is one of a general class of Theorems called *pumping theorems*. They are called *pumping theorems* because they assert the existence of certain points in certain strings where a substring can be repeatedly inserted (pumping) without affecting the acceptability of the string.

We present here two versions of the **Pumping Lemma**. First is the Lecture Notes version from the first edition of the Book and the second is the Book version (page 88) from the new edition.

The Book version is a slight *generalization* of the Lecture version.
Pumping Lemma

Pumping Lemma 1
Let $L$ be an infinite regular language over $\Sigma \neq \emptyset$
Then there are strings $x, y, z \in \Sigma^*$ such that

$$y \neq e \quad \text{and} \quad xy^n z \in L \quad \text{for all} \quad n \geq 0$$

Observe that the Pumping Lemma 1 says that in an infinite regular language $L$, there is a word $w \in L$ that can be re-written as $w = xyz$ in such a way that $y \neq e$ and we "pump" the part $y$ any number of times and still have that such obtained word is still in $L$, i.e. that $xy^n z \in L$ for all $n \geq 0$
Hence the name Pumping Lemma
Role of Pumping Lemma

We use the **Pumping Lemma** as a *tool* to carry *proofs* that some languages **are not regular**

**Proof METHOD**

Given an infinite language $L$ we want to **PROVE** it to be **NOT REGULAR**

We proceed as follows

1. We assume that $L$ is **REGULAR**

2. Hence by **Pumping Lemma** we get that there is a word $w \in L$ that can be **re-written** as $w = xyz$, $y \neq e$, and $xy^nz \in L$ for all $n \geq 0$

3. We examine the fact $xy^nz \in L$ for all $n \geq 0$

4. If we get a **CONTRADICTION** we have proved that the language $L$ is **not regular**
Proof of Pumping Lemma

**Pumping Lemma 1**

Let \( L \) be an infinite regular language over \( \Sigma \neq \emptyset \)

Then there are strings \( x, y, z \in \Sigma^* \) such that

\[
y \neq e \quad \text{and} \quad xy^n z \in L \quad \text{for all} \quad n \geq 0
\]

**Proof**

Since \( L \) is regular, \( L \) is accepted by a deterministic finite automaton

\[
M = (K, \Sigma, \delta, s, F)
\]

Suppose that \( M \) has \( n \) states, i.e. \( |K| = n \) for \( n \geq 1 \)

Since \( L \) is infinite, \( M \) accepts some string \( w \in L \) of length \( n \) or greater, i.e.

there is \( w \in L \) such that \( |w| = k > n \) and

\[
w = \sigma_1 \sigma_2 \ldots \sigma_k \quad \text{for} \quad \sigma_i \in \Sigma, \quad 1 = 1, 2, \ldots, k
\]
Proof of Pumping Lemma

Consider a **computation** of $w = \sigma_1\sigma_2\ldots\sigma_k \in L$:

$$(q_0, \sigma_1\sigma_2\ldots\sigma_k) \vdash_M (q_1, \sigma_2\ldots\sigma_k), \vdash_M (q_2, \sigma_3\ldots\sigma_k), \vdash_M (q_3, \sigma_4\ldots\sigma_k), \ldots, \vdash_M (q_{k-1}, \sigma_k), \vdash_M (q_k, e)$$

where $q_0$ is the initial state $s$ of $M$ and $q_k$ is a final state of $M$.

Since $|lw| = k > n$ and $M$ has only $n$ states, by **Pigeon Hole Principle** we have that there exist $i$ and $j$, $0 \leq i < j \leq k$, such that $q_i = q_j$.

That is, the string $\sigma_{i+1}\ldots\sigma_j$ is nonempty since $i + 1 \leq j$ and drives $M$ from state $q_i$ back to state $q_i$.

But then this string $\sigma_{i+1}\ldots\sigma_j$ could be **removed** from $w$, or we could **insert** any number of its **repetitions** just after just after $\sigma_j$ and $M$ would still **accept** such string.
Proof of Pumping Lemma

We just showed by Pigeon Hole Principle we have that $M$ that accepts $w = \sigma_1\sigma_2\ldots\sigma_k \in L$ also accepts the string $\sigma_1\sigma_2\ldots\sigma_i(\sigma_{i+1}\ldots\sigma_j)^n\sigma_{j+1}\ldots\sigma_k$ for each $n \geq 0$

Observe that $\sigma_{i+1}\ldots\sigma_j$ is non-empty string since $i + 1 \leq j$

That means that there exist strings $x = \sigma_1\sigma_2\ldots\sigma_i$, $y = \sigma_{i+1}\ldots\sigma_j$, $z = \sigma_{j+1}\ldots\sigma_k$ for $y \neq e$

such that $y \neq e$ and $xy^n z \in L$ for all $n \geq 0$
Proof of Pumping Lemma

The computation of $M$ that accepts $xy^n z$ is as follows

$$(q_0, xy^n z) \vdash_M^* (q_i, y^n z ) \vdash_M^* (q_i, y^{n-1} z )$$

$$\vdash_M^* \ldots \vdash_M^* (q_i, y^{n-1} z ) \vdash_M^* (q_k, e)$$

This ends the proof.

Observe that the proof of the holds for for any word $w \in L$ with $|w| \geq n$, where $n$ is the number of states of deterministic $M$ that accepts $L$.

We get hence another version of the **Pumping Lemma 1**
Proof of Pumping Lemma

Pumping Lemma 2
Let $L$ be an infinite regular language over $\Sigma \neq \emptyset$
Then there is an integer $n \geq 1$ such that for any word $w \in L$ with lengths greater than $n$, i.e. $|w| \geq n$ there are $x, y, z \in \Sigma^*$ such that $w$ can be re-written as $w = xyz$ and

$$y \neq e \quad \text{and} \quad xy^n z \in L \quad \text{for all} \quad n \geq 0$$

Proof
Since $L$ is regular, it is accepted by a deterministic finite automaton $M$ that has $n \geq 1$ states
This is our integer $n \geq 1$
Let $w$ be any word in $L$ such that $|w| \geq n$
Such words exist as $L$ in infinite
The rest of the proof exactly the same as in case of Pumping Lemma 1
Pumping Lemma

We write the **Pumping Lemma 2** symbolically using quantifiers symbols as follows

**Pumping Lemma 2**

Let $L$ be an **infinite regular** language over $\Sigma \neq \emptyset$

Then the following holds

$$\exists n \geq 1 \forall w \in L (|w| \geq n \implies \exists x, y, z \in \Sigma^* (w = xyz \land y \neq e \land \forall n \geq 0 (xy^n z \in L)))$$
Book Pumping Lemma

Book Pumping Lemma is a STRONGER version of the Pumping Lemma 2

It applies to any any regular language, not to an infinite regular language, as the Pumping Lemmas 1, 2
Book Pumping Lemma

Let $L$ be a regular language over $\Sigma \neq \emptyset$

Then there is an integer $n \geq 1$ such that any word $w \in L$ with $|w| \geq n$ can be re-written as $w = xyz$ such that

$y \neq \epsilon, \ |xy| \leq n, \ x, y, z \in \Sigma^*$ and $xy^i z \in L$ for all $i \geq 0$

Proof The proof goes exactly as in the case of Pumping Lemmas 1, 2

Notice that from the proof of Pumping Lemma 1

$$x = \sigma_1 \sigma_2 \ldots \sigma_i, \quad z = \sigma_{j+1} \ldots \sigma_k$$ for $0 \leq i < j \leq n$

and so by definition $|xy| \leq n$ for $n$ being the number of states of the deterministic $M$ that accepts $L$
We write the Pumping Lemma symbolically using quantifiers symbols as follows:

**Book Pumping Lemma**

Let $L$ be a regular language over $\Sigma \neq \emptyset$

Then the following holds:

$$\exists n \geq 1 \forall w \in L \left( |w| \geq n \Rightarrow \exists x, y, z \in \Sigma^* (w = xyz \land y \neq e \land |xy| \leq n \land \forall i \geq 0 (xy^iz \in L)) \right)$$

A natural question arises:

WHY the Book Pumping Lemma applies when $L$ is a regular finite language?

When $L$ is a regular finite language the Lecture Lemma does not apply.
Let’s look at an example of a finite, and hence a regular language

\[ L = \{a, b, ab, bb\} \]

**Observe** that the condition

\[ \exists_{n \geq 1} \forall_{w \in L} \ (|w| \geq n) \Rightarrow \exists_{x, y, z \in \Sigma^*}(w = xyz \land y \neq e \land |xy| \leq n \land \forall_{i \geq 0}(xy^iz \in L)) \]

of the **Book Pumping Lemma** holds because there exists \( n = 3 \) such that the conditions becomes as follows
Book Pumping Lemma

Take \( n = 3 \), or any \( n \geq 3 \) we get statement:

\[
\exists_{n=3} \forall_{w \in L} \ (|w| \geq 3 \ \Rightarrow \ \exists_{x,y,z \in \Sigma^*} \ (w = xyz \ \cap \ y \neq e \ \cap \ |xy| \leq n \ \cap \ \forall_{i \geq 0} \ (xy^iz \in L)))
\]

Observe that the above is a TRUE statement because the statement \(|w| \geq 3\) is FALSE for all \( w \in L = \{a, b, ab, bb\} \)

By definition, the implication \( \text{FALSE} \ \Rightarrow \ (\text{anything}) \) is always TRUE, hence the whole statement is TRUE
The same reasoning applies for any finite (and hence regular) language.

**In general,** let $L$ be any finite language.

Let $m = \max\{|w| : w \in L\}$

Such $m$ exists because $L$ is finite.

Take $n = m + 1$ as the $n$ in the condition of the Book Pumping Lemma.

The Lemma condition is TRUE for all $w \in L$, because the statement $|w| \geq m + 1$ is FALSE for all $w \in L$.

By definition, the implication $FALSE \Rightarrow (anything)$ is always TRUE, hence the whole statement is TRUE.
Pumping Lemma Applications

Use **Pumping Lemma** to **prove** the following

**Fact 1**
The language \( L \subseteq \{a, b\}^* \) defined as follows

\[
L = \{a^n b^n : \ n > 0\}
\]

is **NOT** regular.

Obviously, \( L \) is infinite and we use the Lecture version

**Pumping Lemma 1**
Let \( L \) be an **infinite regular** language over \( \Sigma \neq \emptyset \)

Then **there are** strings \( x, y, z \in \Sigma^* \) such that

\[
y \neq e \quad \text{and} \quad xy^n z \in L \quad \text{for all} \quad n \geq 0
\]
Pumping Lemma Applications

Reminder: we proceed as follows
1. We assume that $L$ is REGULAR
2. Hence by Pumping Lemma we get that there is a word $w \in L$ that can be re-written as $w = xyz$ for $y \neq e$ and $xy^n z \in L$ for all $n \geq 0$
3. We examine the fact $xy^n z \in L$ for all $n \geq 0$
4. If we get a CONTRADICTION we have proved that $L$ is NOT REGULAR
Pumping Lemma Applications

Assume that

\[ L = \{a^m b^m : m \geq 0\} \]

IS REGULAR

L is infinite hence Pumping Lemma 1 applies, so there is a word \( w \in L \) that can be re-written as \( w = xyz \) for \( y \neq e \) and \( xy^n z \in L \) for all \( n \geq 0 \)

There are three possibilities for \( y \neq e \)

We will show that in each case we prove that \( xy^n z \in L \) is impossible (contradiction)
Pumping Lemma Applications

Consider \( w = xyz \in L \), i.e. \( xyz = a^m b^m \) for some \( m \geq 0 \)
We have to consider the following cases

**Case 1**
y consists entirely of a’s

**Case 2**
y consists entirely of b’s

**Case 3**
y contains both some a’s followed by some b’s

We will show that in each case assumption that \( xy^n z \in L \) for all \( n \) leads to **CONTRADICTION**
Pumping Lemma Applications

Consider $w = xyz \in L$, i.e. $xyz = a^m b^m$ for some $m \geq 0$

**Case 1:** $y$ consists entirely of $a$’s
So $x$ must consist entirely of $a$’s only and $z$ must consist of some $a$’s followed by some $b$’s.
Remember that only we must have that $y \neq e$
We have the following situation

$x = a^p$ for $p \geq 0$ as $x$ can be empty
$y = a^q$ for $q > 0$ as $y$ must be nonempty
$z = a^r b^s$ for $r \geq 0, s > 0$ as we must have some $b$’s
Pumping Lemma Applications

The condition \( xy^n z \in L \) for all \( n \geq 0 \) becomes as follows

\[
a^p(a^q)^n a^r b^s = a^{p+nq+r} b^s \in L
\]

for all \( p, q, n, r, s \) such that the following conditions hold

\( C1: \quad p \geq 0, \quad q > 0, \quad n \geq 0, \quad r \geq 0, \quad s > 0 \)

By definition of \( L \)

\[
a^{p+nq+r} b^s \in L \quad \text{iff} \quad [p + nq + r = s]
\]

Take case: \( p = 0, \quad r = 0, \quad q > 0, \quad n = 0 \)

We get \( s = 0 \quad \text{CONTRADICTION} \quad \text{with} \ C1: \quad s > 0 \)
Pumping Lemma Applications

Consider $xyz = a^m b^m$ for some $m \geq 0$

**Case 2:** $y$ consists of $b$’s only
So $x$ **must** consists of some $a$’s followed by some $b$’s and $z$ **must** have only $b$’s, possibly none
We have the following situation

\[ x = a^p b^r \quad \text{for} \quad p > 0 \quad \text{as} \quad y \quad \text{has at least one} \quad b \quad \text{and} \quad r \geq 0 \]

\[ y = b^q \quad \text{for} \quad q > 0 \quad \text{as} \quad y \quad \text{must be nonempty} \]

\[ z = b^s \quad \text{for} \quad s \geq 0 \]
Pumping Lemma Applications

The condition $xy^nz \in L$ for all $n \geq 0$ becomes as follows

$$a^pb^r(b^q)^nb^s = a^pb^{r+nq+r} \in L$$

for all $p, q, n, r, s$ such that the following conditions hold

**C2:** $p > 0, \ r \geq 0, \ q > 0, \ n \geq 0, \ s \geq 0$

By definition of $L$

$$a^pb^{r+nq+r} \in L \iff [p = r + qn + s]$$

Take case: $r = 0, \ n = 0, \ q > 0$

We get $p = 0$ CONTRADICTION with **C2:** $p > 0$
Pumping Lemma Applications

Consider $xyz = a^m b^m$ for some $m \geq 0$

**Case 3:** $y$ contains both $a$’s and $a$’s
So $y = a^p b^r$ for $p > 0$ and $r > 0$
Case $y = b^r a^p$ is impossible
Take case: $y = ab$, $x = e$, $z = e$ and $n = 2$
By Pumping Lemma we get that $y^2 \in L$
But this is a CONTRADICTION with $y^2 = abab \notin L$
We covered all cases and it ends the proof
Pumping Lemma Applications

Use Pumping Lemma to prove the following
Fact 2
The language $L \subseteq \{a\}^*$ defined as follows

$$L = \{a^n : n \in \text{Prime}\}$$

IS NOT regular

Obviously, $L$ is infinite and we use the Lecture version

Proof

Assume that $L$ is regular, hence as $L$ is infinite, so there is a word $w \in L$ that can be re-written as $w = xyz$ for $y \neq e$ and $xy^nz \in L$ for all $n \geq 0$

Consider $w = xyz \in L$, i.e. $xyz = a^m$ for some $m > 0$ and $m \in \text{Prime}$
Pumping Lemma Applications

Then

\[ x = a^p, \quad y = a^q, \quad z = a^r \quad \text{for} \quad p \geq 0, \quad q > 0, \quad r \geq 0 \]

The condition \( xy^n z \in L \) for all \( n \geq 0 \) becomes as follows

\[ a^p (a^q)^n a^r = a^{p+nq+r} \in L \]

It means that for all \( n, p, q, r \) the following condition hold

\[ \text{C} \quad n \geq 0, \quad p \geq 0, \quad q > 0, \quad r \geq 0, \quad \text{and} \quad p + nq + r \in \text{Prime} \]

But this is IMPOSSIBLE
Pumping Lemma Applications

Take $n = p + 2q + r + 2$ and evaluate:

$p + nq + r = p + (p + 2q + r + 2)q + r =

p(1 + q) + 2q(q + 1) + r(q + 1) = (q + 1)(p + 2q + r)$

By the above and the condition $C$ we get that

$p + nq + r \in \text{Prime}$ and $p + nq + r = (q + 1)(p + 2q + r)$

and both factors are natural numbers greater than 1 what is a CONTRADICTION

This ends the proof