Lecture Outline

Nonregular Languages
- Nonregular Languages
- Pumping Lemma
- Examples of nonregular languages

Examples

For a fixed integer $n \geq 0$, define a language $L_n = \{0^n 1^n\}$.
Q: Is L_n regular?

Examples

For a fixed integer $n \geq 0$, define a language $L_n = \{0^n 1^n\}$.
Q: Is L_n regular?

Q: Are the following languages regular?

$$L_0 \cup L_1, \quad L_0 \cup L_1 \cup L_2, \quad L_0 \cup L_1 \cup L_2 \cup L_3$$

Examples

For a fixed integer $n \geq 0$, define a language $L_n = \{0^n 1^n\}$.
Q: Is L_n regular?

Q: Are the following languages regular?

$$L_0 \cup L_1, \quad L_0 \cup L_1 \cup L_2, \quad L_0 \cup L_1 \cup L_2 \cup L_3$$

Q: Is the finite union $L_0 \cup L_1 \cup \cdots \cup L_m$ regular (for a *fixed* integer $m \geq 0$)?

Examples

For a fixed integer $n \geq 0$, define a language $L_n = \{0^n 1^n\}$.
Q: Is L_n regular?

Q: Are the following languages regular?

$$L_0 \cup L_1, \quad L_0 \cup L_1 \cup L_2, \quad L_0 \cup L_1 \cup L_2 \cup L_3$$

Q: Is the finite union $L_0 \cup L_1 \cup \cdots \cup L_m$ regular (for a *fixed* integer $m \geq 0$)?

Q: Is the infinite union of the languages L_n regular?

$$\bigcup_{n \geq 0} = L_0 \cup L_1 \cup \cdots = \{0^n 1^n \mid n \geq 0\}.$$

Nonregular Languages

Some languages seem to require an infinite number of states to be recognized by NFA.

$$C = \{ w \mid w \text{ has the same number of 0's and 1's} \}$$

$$D = \{ w \mid w \text{ has the same number substrings 01 and 10} \}$$

Nonregular Languages

Some languages seem to require an infinite number of states to be recognized by NFA.

$$C = \{w \mid w \text{ has the same number of 0's and 1's}\}$$

$$D = \{w \mid w \text{ has the same number substrings 01 and 10}\}$$

However, only those that *cannot* be recognized with any finite number of states (recall that F in DFA/NFA stands for "finite") are nonregular.
Surprisingly, the language D can be recognized by an NFA, i.e., D is regular. The language C is not but that requires a proof.

Proving nonregularness

Q: How one can prove that a language is regular?

Proving nonregularness

Q: How one can prove that a language is regular?

Q: How one can prove that a language is NOT regular?

Proving nonregularness

Q: How one can prove that a language is regular?

Q: How one can prove that a language is NOT regular?

We need to find a property that (i) characterizes regular languages; and (ii) can be more or less easily tested for a given language.

Proving nonregularness

Q: How one can prove that a language is regular?

Q: How one can prove that a language is NOT regular?

We need to find a property that (i) characterizes regular languages; and (ii) can be more or less easily tested for a given language.

Unfortunately, non-existence of an NFA recognizing a given language is not easy to test — we need another property. One such useful property is given by the Pumping Lemma.

Pumping Lemma

Lemma

If A is a regular language, then there exists an integer $p > 0$ (called the pumping length) such that for any string $s \in A$ with $|s| \geq p$, the string s can be divides into three pieces: $s = xyz$, satisfying the following conditions:

1. *for each $i \geq 0$, $xy^iz \in A$;*
2. $|y| > 0$*; and*
3. $|xy| \leq p$.

Q: What would be a proof outline?

Pumping Lemma

Lemma

If A is a regular language, then there exists an integer $p > 0$ (called the pumping length) such that for any string $s \in A$ with $|s| \geq p$, the string s can be divides into three pieces: $s = xyz$, satisfying the following conditions:

1. *for each $i \geq 0$, $xy^iz \in A$;*
2. *$|y| > 0$; and*
3. *$|xy| \leq p$.*

Q: What would be a proof outline?

We will prove the pumping lemma by construction of p, x, y, and z for a given element $s \in A$ and a DFA N recognizing A, and show that they satisfy conditions 1, 2, and 3.

Proof of the Pumping Lemma: Construction

For a given regular language A, consider a DFA $N = (Q, \Sigma, q_0, \delta, F)$ that recognizes A.

Proof of the Pumping Lemma: Construction

For a given regular language A, consider a DFA $N = (Q, \Sigma, q_0, \delta, F)$ that recognizes A.

Let $p = |Q|$, i.e., the number of states in N. For a given string $s \in A$ of length $|s| = m \geq p$, consider a computation $(r_0, r_1, \ldots, r_m)$ of N that accepts s, i.e.:

$$r_0 \xrightarrow{s_1} r_1 \xrightarrow{s_2} r_2 \xrightarrow{s_3} \cdots \xrightarrow{s_{m-1}} r_{m-1} \xrightarrow{s_m} r_m$$

where $s_1 s_2 \ldots s_m = s$, $r_0 = q_0$, and $r_m \in F$.

Proof of the Pumping Lemma: Construction

For a given regular language A, consider a DFA $N = (Q, \Sigma, q_0, \delta, F)$ that recognizes A.

Let $p = |Q|$, i.e., the number of states in N. For a given string $s \in A$ of length $|s| = m \geq p$, consider a computation $(r_0, r_1, \ldots, r_m)$ of N that accepts s, i.e.:

$$r_0 \xrightarrow{s_1} r_1 \xrightarrow{s_2} r_2 \xrightarrow{s_3} \cdots \xrightarrow{s_{m-1}} r_{m-1} \xrightarrow{s_m} r_m$$

where $s_1 s_2 \ldots s_m = s$, $r_0 = q_0$, and $r_m \in F$.

By the *pigeonhole principle*, at least two states among $r_0, r_1, \ldots, r_p$ must be equal, i.e., $r_i = r_j$ for some i, j such that $0 \leq i < j \leq p$.

Proof of the Pumping Lemma: Construction

For a given regular language A, consider a DFA $N = (Q, \Sigma, q_0, \delta, F)$ that recognizes A.

Let $p = |Q|$, i.e., the number of states in N. For a given string $s \in A$ of length $|s| = m \geq p$, consider a computation $(r_0, r_1, \ldots, r_m)$ of N that accepts s, i.e.:

$$r_0 \xrightarrow{s_1} r_1 \xrightarrow{s_2} r_2 \xrightarrow{s_3} \cdots \xrightarrow{s_{m-1}} r_{m-1} \xrightarrow{s_m} r_m$$

where $s_1 s_2 \ldots s_m = s$, $r_0 = q_0$, and $r_m \in F$.

By the *pigeonhole principle*, at least two states among $r_0, r_1, \ldots, r_p$ must be equal, i.e., $r_i = r_j$ for some i, j such that $0 \leq i < j \leq p$.

We define

$$x = s_1 s_2 \ldots s_i, \quad y = s_{i+1} s_{i+2} \ldots s_j, \quad z = s_{j+1} s_{j+2} \ldots s_m.$$

Proof of the Pumping Lemma: Verification

We need to show that the defined p, x, y, and z satisfy the statement of the Lemma.

First off, we have by constructions:

$$xyz = s_1 s_2 \ldots s_i \; s_{i+1} s_{i+2} \ldots s_j \; s_{j+1} s_{j+2} \ldots s_m = s.$$

Proof of the Pumping Lemma: Verification

We need to show that the defined p, x, y, and z satisfy the statement of the Lemma.

First off, we have by constructions:

$$xyz = s_1 s_2 \ldots s_i \; s_{i+1} s_{i+2} \ldots s_j \; s_{j+1} s_{j+2} \ldots s_m = s.$$

We need to show that
1. for each $i \geq 0$, $xy^i z \in A$
Q: Something is wrong with this statement, isn't it?

Proof of the Pumping Lemma: Verification

We need to show that the defined p, x, y, and z satisfy the statement of the Lemma.

First off, we have by constructions:

$$xyz = s_1 s_2 \dots s_i \, s_{i+1} s_{i+2} \dots s_j \, s_{j+1} s_{j+2} \dots s_m = s.$$

We need to show that
1. for each $i \geq 0$, $xy^i z \in A$
Q: Something is wrong with this statement, isn't it?

To avoid confusion with i defined in our proof, let us use different symbol in the condition 1:
1. for each $k \geq 0$, $xy^k z \in A$
Q: How to prove that this condition holds?

Proof of the Pumping Lemma: Verification (cont'd)

From a computation for $s = xyz$ on N, we will construct a computation for xy^kz on N for any integer $k \geq 0$.

Here is the computation for s where the equal states r_i and r_j denoted by R:

$$r_0 \xrightarrow{s_1} r_1 \ldots r_{i-1} \xrightarrow{s_i} (R \xrightarrow{s_{i+1}} r_{i+1} \ldots r_{j-1} \xrightarrow{s_j}) R \xrightarrow{s_{j+1}} q_{j+1} \ldots r_{m-1} \xrightarrow{s_m} r_m$$

Note that the part in the parentheses can be repeated k times in a row, using the same transitions multiple times. For example, for $k = 2$, we will have in the middle

$$\ldots r_{i-1} \xrightarrow{s_i} (R \xrightarrow{s_{i+1}} r_{i+1} \ldots r_{j-1} \xrightarrow{s_j})(R \xrightarrow{s_{i+1}} r_{i+1} \ldots r_{j-1} \xrightarrow{s_j}) R \xrightarrow{s_{j+1}} q_{j+1} \ldots$$

This way we can construct a computation for xy^kz that have the same starting and ending states as in the computation for s, and using the same transitions but possibly multiple times.

The actual proof is done by induction on k.

Proof of the Pumping Lemma: Verification (cont'd)

Now let us prove the second condition:

2. $|y| > 0$;

Recall that $y = s_{i+1}s_{i+2}\ldots s_j$ for some i, j such that $0 \leq i < j \leq p$.

Proof of the Pumping Lemma: Verification (cont'd)

Now let us prove the second condition:

2. $|y| > 0$;

Recall that $y = s_{i+1}s_{i+2}\dots s_j$ for some i, j such that $0 \leq i < j \leq p$.

Finally, we need to prove the third condition:

3. $|xy| \leq p$.

Recall that $x = s_1s_2\dots s_i$ so that $xy = s_1s_2\dots s_is_{i+1}s_{i+2}\dots s_j$.

Proving Nonregularness with Pumping Lemma

The following corollary of Pumping Lemma is often used for proving that a given language is nonregular.

Corollary

A language A is nonregular if for every integer $p > 0$, there exists a string $s \in A$ with $|s| \geq p$ that cannot be divided into three pieces: $s = xyz$, satisfying the following conditions:

1. *for each $i \geq 0$, $xy^iz \in A$;*
2. $|y| > 0$*; and*
3. $|xy| \leq p$*.*

Examples of nonregular languages

Prove that the following languages are not regular:

$B = \{0^n 1^n \mid n \geq 0\}$

Examples of nonregular languages

Prove that the following languages are not regular:

$B = \{0^n 1^n \mid n \geq 0\}$

$F = \{ww \mid w \in \{0,1\}^*\}$

Examples of nonregular languages

Prove that the following languages are not regular:

$$B = \{0^n 1^n \mid n \geq 0\}$$

$$F = \{ww \mid w \in \{0,1\}^*\}$$

$$D = \{1^{n^2} \mid n \geq 0\}$$

Examples of nonregular languages

Prove that the following languages are not regular:

$B = \{0^n 1^n \mid n \geq 0\}$

$F = \{ww \mid w \in \{0,1\}^*\}$

$D = \{1^{n^2} \mid n \geq 0\}$

$E = \{0^i 1^j \mid i > j\}$