Notes

Chapter #1

Discrete Mathematics

(9965864/س) 

(م)
Chapter 1: Formal Logic

In this chapter we will talk about formal (aka, symbolic) logic & its applications in computer science.

But, first we must find a notation to represent arguments in a formal way.

**Idea:** Represent argument by symbols to make them meaningless.

**Why?**
To allow us to concentrate on the pure form of argument.

Statements are sentences with a truth value.

Simple sentence can be represented by letter (Statement letter, A, B, ... etc), then these letters can be combined to form another sentence.
Logical connectives between statements:

1. Conjunction
2. Disjunction
3. Implication
4. Equivalence
5. Negation.

We will talk about that later.

Sentences:

Examples #1

A: New York is a city in USA.
B: Kuwait is Part of Europe.
C: 1+1 = 2
D: 2+2 = 3

A, B, C, D are statements, as we can determine if they are True or False.

\[
\begin{align*}
\Rightarrow & \text{ A \& C are True statements.} \\
\Rightarrow & \text{ B \& D are False statements.}
\end{align*}
\]
Examples #2

A: What time is it?
B: Read this carefully.
C: \( x + 1 = 2 \)
D: \( x + y = 2 \)

A, B, C, D are not statements, because
(A) is a question, (B) is a demand
(C, D) can not be determined if T or F

\[ T \leftrightarrow False \]

So, and because we can't determine
if they are T or F, they are not

Statements

BUT, in C & D can be turned into
statement if variables values determined.
Exercise:

Determine if the following sentences are statements (propositions)? and if they are determine if T or F.

⇒ A: The population of Kuwait is 100 m.  
   This is a statement, and it is False.

⇒ B: January has 31 days.  
   This is a statement, and it is True.

⇒ C: 2 + 3 = 5  
   Statement, True.

⇒ D: 5 + 7 = 10  
   Statement, False.

⇒ E: Answer this question.  
   Not a statement.

⇒ F: x + 2 = 11  
   Non-statement, until know (x)

⇒ G: The moon is made of green cheese.  
   Statement, False
**Compound propositions:**

We can combine one or more statement together by using "logical connectives".

For each connectivity, we will state the "truth table", which lists the truth value of the compound proposition for all possible values of its variables.

**Conjunction:** which means **AND**

Let \( A, B \) be statements, conjunction of them is "\( A \land B \)" or "\( A \) and \( B \)". It is true only when \( A, B \) are true and false otherwise.

<table>
<thead>
<tr>
<th></th>
<th></th>
<th>( A \land B )</th>
</tr>
</thead>
<tbody>
<tr>
<td>T</td>
<td>T</td>
<td>T</td>
</tr>
<tr>
<td>T</td>
<td>F</td>
<td>F</td>
</tr>
<tr>
<td>F</td>
<td>T</td>
<td>F</td>
</tr>
<tr>
<td>F</td>
<td>F</td>
<td>F</td>
</tr>
</tbody>
</table>
**Disjunction:** which means **OR**

Let \( A, B \) be statements, disjunction of them is " \( A \lor B \) " or "A or B"

It is false only when \( A, B \) are false and true otherwise

<p>| | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>T</td>
<td>T</td>
<td>T</td>
</tr>
<tr>
<td>T</td>
<td>F</td>
<td>T</td>
</tr>
<tr>
<td>F</td>
<td>T</td>
<td>T</td>
</tr>
<tr>
<td>F</td>
<td>F</td>
<td>F</td>
</tr>
</tbody>
</table>

**Implication:**

Let \( A, B \) statements, conditional statement is " \( A \rightarrow B \) " or " if A, then B"

A is called hypothesis

B is called conclusion

(False only if A is True and B is False)
Implication Examples:

A: If the sun is up, then it is day
   
   hypothesis

   conclusion

B: If \( x = 2 \), then \( x + 3 = 5 \)
   
   hypothesis

   conclusion

Equivalence:

let \( A, B \) statements, equivalence is “\( A \leftrightarrow B \)”, or “\( A \) if and only if \( B \)”

True if both values has same truth value.

<table>
<thead>
<tr>
<th>A</th>
<th>B</th>
<th>A ( \leftrightarrow ) B</th>
</tr>
</thead>
<tbody>
<tr>
<td>T</td>
<td>T</td>
<td>T</td>
</tr>
<tr>
<td>T</td>
<td>F</td>
<td>F</td>
</tr>
<tr>
<td>F</td>
<td>T</td>
<td>F</td>
</tr>
<tr>
<td>F</td>
<td>F</td>
<td>T</td>
</tr>
</tbody>
</table>
Negation:

The connectivities 1 to 4 called binary as they have two sides A & B.

But this called "unary" as it has only one side.

Let A is statement, negation is "A" or "not A" or "\( \neg A \)"

It has the opposite value of A

<table>
<thead>
<tr>
<th>A</th>
<th>A'</th>
</tr>
</thead>
<tbody>
<tr>
<td>T</td>
<td>F</td>
</tr>
<tr>
<td>F</td>
<td>T</td>
</tr>
</tbody>
</table>
Exercise: Name the antecedent (hypothesis) and consequent (conclusion) in following:

⇒ If the rain continues, then the river will flood.

\[ \text{Sol} \rightarrow A: \text{The rain continues.} \]
\[ B: \text{The river will flood.} \]

⇒ A sufficient condition for network failure is that central switch goes down.

\[ \text{Sol} \rightarrow A: \text{Central switch goes down.} \]
\[ B: \text{Network fail} \]

⇒ A good diet is a necessary condition for a healthy cat.

\[ \text{Sol} \rightarrow A: \text{Good diet.} \]
\[ B: \text{Healthy cat.} \]

⇒ Healthy planet growth follows from sufficient water

\[ \text{Sol} \rightarrow A: \text{Sufficient Water} \]
\[ B: \text{Healthy planet growth} \]
Exercise:

Given that \( A \) is True, \( B \) is False, \( C \) is True.
Find Truth value?

\[ \Rightarrow (A \land B) \lor C \]

Sol.

\[ (T \land F) \lor T \]

\[ F \lor T = \boxed{\text{T}} \]

\[ \Rightarrow (A \rightarrow C) \land B \]

Sol.

\[ (T \rightarrow T) \land F \]

\[ T \land F = \boxed{\text{F}} \]

\[ \Rightarrow (A \land B) \rightarrow C \]

Sol.

\[ (T \land F) \rightarrow T \]

\[ F \rightarrow T = \boxed{\text{T}} \]

\[ \Rightarrow (A \rightarrow B) \land (B \rightarrow A) \]

Sol.

\[ (T \rightarrow F) \land (F \rightarrow T) \]

\[ F \land T = \boxed{\text{F}} \]
Negation Exercise:

1. Assume A: "Julie likes butter but hates cream"

   What is \( A' \)?

   a) Julie hates butter and cream.

   b) Julie does not like butter or cream.

   \( A' \leftarrow \{c\) Julie dislikes butter but loves cream.

   d) Julie hates butter or likes cream.

2. Write a statement \( A' \), if A is
   "John is either tall or fat"

   \( \to A' \): John is neither tall nor fat
Well-Formed Formula (wff):

Any string is a group of letters, connectives and parenthesis together to form an expression.

Example:

\[(A \rightarrow B)^n (B \rightarrow A) \checkmark\]

It is a correct expression.

\[A\] \checkmark \rightarrow BC \times

It is not legitimate (legal).

So we need to know the order of precedence of connectives.

<table>
<thead>
<tr>
<th>1 \rightarrow ( )</th>
<th>parantheses</th>
</tr>
</thead>
<tbody>
<tr>
<td>2 \rightarrow ', \neg</td>
<td>negation</td>
</tr>
<tr>
<td>3 \rightarrow \land, \lor</td>
<td>And, OR</td>
</tr>
<tr>
<td>4 \rightarrow \rightarrow</td>
<td>if, then</td>
</tr>
<tr>
<td>5 \rightarrow \leftrightarrow</td>
<td>equal</td>
</tr>
</tbody>
</table>
Tautologies vs. Contradiction:

is the wff whose truth values are always true.

Example:

$1 + 1 = 2$ (is odd)  
(A ∨ B)

Example:

$1 + 2 = 5$  
(A ∧ B)

Example: Construct the truth table for

$\Rightarrow (A ∨ A') → (B ∧ B')$

<table>
<thead>
<tr>
<th>A</th>
<th>B</th>
<th>A'</th>
<th>B'</th>
<th>A ∨ A'</th>
<th>B ∧ B'</th>
<th>(A ∨ A') → (B ∧ B')</th>
</tr>
</thead>
<tbody>
<tr>
<td>T</td>
<td>T</td>
<td>F</td>
<td>F</td>
<td>T</td>
<td>F</td>
<td>F</td>
</tr>
<tr>
<td>T</td>
<td>F</td>
<td>F</td>
<td>F</td>
<td>T</td>
<td>F</td>
<td>F</td>
</tr>
<tr>
<td>F</td>
<td>T</td>
<td>T</td>
<td>F</td>
<td>F</td>
<td>F</td>
<td>F</td>
</tr>
<tr>
<td>F</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>T</td>
<td>F</td>
<td>F</td>
</tr>
</tbody>
</table>

So $(A ∨ A') → (B ∧ B')$ is a Contradiction
Fautological equivalences:

**Boolean Algebra**

1. **General**

   - \( A \lor B \iff B \lor A \)
   - \( A \land B \iff B \land A \)
   - \( (A \lor B) \land C \iff A \land (B \land C) \)
   - \( (A \land B) \lor C \iff A \lor (B \lor C) \)
   - \( A \lor 0 \iff A \)
   - \( A \land 1 \iff 1 \)
   - \( A \land A' \iff 0 \)
   - \( A \lor A' \iff 1 \)
   - \( A' \iff A \)

2. **De Morgan’s Laws**:

   - \( (A \lor B)' \iff (A' \land B') \)
   - \( (A \land B)' \iff (A' \lor B') \)

3. **Implication & Equivalence**:

   - \( (P \rightarrow Q) \iff (P' \lor Q) \)
   - \( (P \leftrightarrow Q) \iff (P \rightarrow Q) \land (Q \rightarrow P) \)
Arguments:

Argument is like implication, but with a group of statements as a hypothesis and only one statement as a conclusion.

\[ P_1 \land P_2 \land \cdots \land P_n \rightarrow Q \]

given hypotheses

Conclusion

\[ \Rightarrow \text{Valid Arguments:} \]

\[ P_1 \land P_2 \land \cdots \land P_r \rightarrow Q \]

is valid argument when it is a tautology.

Derivation Rules:

To test whether an argument is valid or not, we can use "Proof Sequence" which is a sequence of wffs from the rules of equivalence (Tautological equivalences).
**Derivations Rules : Equivalence :**

<table>
<thead>
<tr>
<th>Expression</th>
<th>Equivalent to</th>
<th>Name / abbr.</th>
</tr>
</thead>
<tbody>
<tr>
<td>( P \lor Q )</td>
<td>( Q \lor P )</td>
<td>Commutative / comm</td>
</tr>
<tr>
<td>( P \land Q )</td>
<td>( Q \land P )</td>
<td></td>
</tr>
<tr>
<td>( P \lor (Q \lor R) )</td>
<td>( (P \lor Q) \lor R )</td>
<td>Associative / assoc</td>
</tr>
<tr>
<td>( P \land (Q \land R) )</td>
<td>( (P \land Q) \land R )</td>
<td></td>
</tr>
<tr>
<td>( (P \lor Q)' )</td>
<td>( P' \lor Q' )</td>
<td>De Morgan's Law / De Morgan</td>
</tr>
<tr>
<td>( (P \land Q)' )</td>
<td>( P' \land Q' )</td>
<td></td>
</tr>
<tr>
<td>( P \rightarrow Q )</td>
<td>( P' \lor Q )</td>
<td>Implication / imp</td>
</tr>
<tr>
<td>( P )</td>
<td>( P'' )</td>
<td>Double negation / dm</td>
</tr>
<tr>
<td>( P \leftrightarrow Q )</td>
<td>( P \leftrightarrow Q )</td>
<td>Equivalence / equ</td>
</tr>
</tbody>
</table>

**Derivation Rules : Inference :**

<table>
<thead>
<tr>
<th>Form</th>
<th>Can derive</th>
<th>Name / abbr.</th>
</tr>
</thead>
<tbody>
<tr>
<td>( P, P \rightarrow Q )</td>
<td>( Q )</td>
<td>Modus ponens / mp</td>
</tr>
<tr>
<td>( Q', P \rightarrow Q )</td>
<td>( P' )</td>
<td>Modus tollens / mt</td>
</tr>
<tr>
<td>( P, Q )</td>
<td>( P \land Q )</td>
<td>Conjunction / con</td>
</tr>
<tr>
<td>( P \land Q )</td>
<td>( P, Q )</td>
<td>Simplification / sim</td>
</tr>
<tr>
<td>( P, Q )</td>
<td>( P \lor Q )</td>
<td>Addition / add</td>
</tr>
</tbody>
</table>
Example:

By using derivation rules, prove that the argument:

\[(A \lor B') \lor B \rightarrow A\]

is valid.

Soli:

1. \[A \lor B'] \quad \text{hyp}\]
2. \[B \quad \text{hyp}\]
3. \[B' \lor A \quad 1, \text{ comm}\]
4. \[B \rightarrow A \quad 3, \text{ imp}\]
5. \[A \quad 2, 4, \text{ mp}\]
Example:-

Prove that argument:

\[ [A \rightarrow (B \lor C)] \land B' \land C' \rightarrow A' \]

is valid.

Sol:-

1. \( A \rightarrow B \lor C \) hyp
2. \( B' \) hyp
3. \( C' \) hyp
4. \( B' \land C' \) 2, 3, Con
5. \( (B \lor C)' \) 4, De Morgan
6. \( A' \) 1, 5, mt
Example:

Prove that argument:

\[ A \land (B \rightarrow C) \land [(A \land B) \rightarrow (D \lor C')] \rightarrow D \]

is valid.

Sol:-

1. A
2. B \rightarrow C
3. (A \land B) \rightarrow (D \lor C')
4. B
5. C
6. A \land B
7. D \lor C'
8. C' \lor D
9. C \rightarrow D
10. D

hyp

2, 4, mp
1, 4, con
3, 6, mp
7, comm
8, imp
5, 9, mp
Deduction method:

Suppose the argument you try to prove has the form:

\[ P_1 \land P_2 \land \cdots \land P_n \rightarrow (R \rightarrow S) \]

We can add \( R \) to the group of hypotheses and derive

\[ P_1 \land P_2 \land \cdots \land P_n \land R \rightarrow S \]

Example:

Prove argument validation

\[(A \rightarrow B) \land (B \rightarrow C) \rightarrow (A \rightarrow C)\]

Solution:

\[ (A \rightarrow B) \land (B \rightarrow C) \land A \rightarrow C \]

1. \( A \rightarrow B \)
2. \( B \rightarrow C \)
3. \( A \)
4. \( B \)  
   hyp
5. \( C \)  
   1, 3, mp

6. \( C \)  
   2, 4, mp