第２回目講義　（10 月 12 日）基本教材のガイドと補足:岡田光弘担当(Hb 版)

以下に項目１－７がありますが、４，５，７は特別に興味がなければスキップしていただいても後の授業に十分ついていくことができます。

リマインダー：基本的教材「計算論理学講義録」をアップしています。毎回のようにこの教材を参照するので、まだダウンロードしていないかたはダウンロードしておいてください。

前回の授業では、「授業形式、課題出題の仕方、成績評価基準について」の確認、学期を通して用いる基本教材の確認をしました。受講に当たってのアンケートに答えていただきました。皆さんそれぞれの履修の動機も分かって大変参考になります。

なお、初回の必須提出物課題（成績評価対象となるもの）は、第５章の内容に入ったところでお出しします。３章、４章と５５章の基本部分を出題対象とします。あくまでも皆さんが教材の基本内容を把握されていることを確かめるための出題ですので、難しくはありません。オーディオ・ビジュアル教材や質問時間も近く導入予定です。

今回は、

１．前回の授業概要の復習

２．前回出した「問い」の答え合わせ（定義 3.1 の論理式の定義（文法）の理解度を測る問いでした）

次に、

３．　基本教材の８ページ後半の、カッコの省略の取り決め

４．評価 S を目標にしている人レベル向けの「中置記法と前置記法について」の補足と問い（提出対象ではありません）

５．最上級者のためのクイズ提出課題（レポート形式で提出できます。必須ではなく、この第１回目講義の内容の先のことを調べてみたい上級者の人へのクイズです。）

６．今週は、第４章「命題論理の意味論」を読んで、例題で解き方を学習していただきます。この章では「真理表」によって論理式（論理言語の「文」）の真偽の値の付け方を学習します。真理表による計算ができるようになれば十分です。

この章の内容は皆さんにとっては簡単だと思います。充分なレベルまで達したかどうかを確認する目的で、例題を見た後、第４章の練習問題のいくつにも取り組んでみてください。提出の必要はありませんが、できる範囲で計算しておいてください。次回の第 3 回講義で重要な練習問題については答え合わせをします。「真理表」の考えは他の科目で学習した方も何人かいると思います。表記法（たとえば真偽を記号で表現するか、１－０という２値の数で表現するかなどの表記上の相違はあるかもしれませんが、どの「真理表」も基本的には

変わりありません。表面的な記法の違いは気にせずに学習してください。
また、練習問題の中には、次の第５章への繋ぎとなるような問題も含まれています。次週はそれらも見ながら、第５章へも入りたく思います。第５章では特に第６章の計算論で用いる論理推論を詳しく見ます。このような流れで今後進みます。

二つの「問い」を加えました。考えてみてください。提出の必要はありませんが、来週の第３回目講義で答え合わせをします。

７．　上級者向けの補足：真理表はとても単純な真偽値の計算ツールですが、この背景にあり、より一般的な意味論の立場である「表示的意味論」の基本的なコンセプトを知っておくことは、特に上級の受講者にとっては有用なことだと言えます。命題論理言語という最も単純な論理言語では「表示的意味論」は単なる「真理表の意味論」でしかないのですが、型月ラムダ計算や述語論理言語なそ、この基本教材でも現れるより進んだ計算言語の意味論の基本となるのが「表示的意味論」のコンセプトです。プログラム意味論や自然言語形式意味論など広い範囲でもこのコンセプトが用いられることが多いです。
そこで、第４章の「真理表」の背景にあるより一般的な「表示的意味論」の導入的開設を上級者向けに行います。特に現時点では上級用開設を必要としない人は、第４章の例題が分かり、練習問題が解ければ、特に以下の、上級者向け「表示的意味論」解説は読まなくても結構です。

１．前回の復習

初回（先週）は基本教材「計算論理学講義ノート」の第３章　（特に第７ページ）でした。。。
（１章、２章も読みたい方は混んでいただいて結構ですが、これらの短い予備的な章はスキップして、直接第７ページの第３章から学習することを注意しました。）

３章の最初にこの「命題論理言語」という記号言語（記号による論理言語なので「形式言語」と呼びこともあります）の単語帳が定義 3.1 として載っています。
ＰやＲは命題変項と呼ばれる単文（接続詞が含まれていない単純文）で、論理記号は接続詞にあたります。
日本語の単文「今日は晴れている」や「太郎は学生である」に対応する単文を表す記号です。
時や場合により「今日は晴れている」や「太郎は学生である」の真偽値は変わるので、」これらは命題変項と呼ばれています。
時や場合で真偽値が変わらない特殊な単文を命題定項と呼びました。例えば、T は常に真を表す単文です。T に対応する日本語の単文は「真である」となります。
単文から論理的接続しを組み合わせてより複雑な複合文を作る文法規則が定義 3.2 の形で

示されています。
命題論理の言語が人工言語としても最も単純であることはこの定義.3.2 で示されている文法規則の数の少なさからもわかります。英語の文法書は何百ページに及ぶ規則がかｋれているでしょうし、日本語でもそうでしょう。一方、プログラミング言語の文法書はずっと短いわけです。しかし、プログラミング言語の文法規則と比べても、命題論理の文法規則の数の少なさは特徴的です。

２　前回の「問い」の答え合わせ

理解できたかどうかを試すのに、次の問題を考えてみていただきました。あくまでも、***この定義 3.2 の文法規則（論理式の生成規則）だけに従って***次の問いに答えていただくという設定でした。

いま単文は、Ｐ、Ｑ、Ｒなどの記号をつかっています。
ここで、最初から抽象的な記号法だけで進めるより、すこし日常の言語の文と関わらせた方が分かりやすいとおもいますので、つぎような対応関係を考えてみましょう。

まず、前回の「問い」を再現します。
今、今夜のパーティのことを話題にしているとします。三密は回避できる環境のパーティだとします。
Ｐは　「太郎が来る」を表す文とします。
Ｏは、「次郎が来る」を表す文とします。
Ｒは、「三郎が来る」を表す文とします。
Ｓは、「花子は嬉しい」を表す文とします。

「かつ」　「または」　「ならば」「…ではない」
に対応する命題論理言語の論理記号は教材７ページ定義 3.1 とその下の日本語との対応表にある通りです。

つぎの日本語の文を、命題論理言語の文（命題論理言語の論理式）に書き直してください。

(i)太郎が来れば花子は嬉しい。
(ii) 太郎も次郎も来れば花子は嬉しくない。
(iii)太郎か来るかまたは次郎が来るかまたは三郎が来るかのいずれかであれば、花子は嬉しい。

(iv)太郎と三郎のうちの一人だけが来れば花子は嬉しい、そして、太郎と三郎の両方が来れば花子は嬉しくない
(v) 三郎が来れば花子が嬉しいというようなこは(決して)ない。

「この問題は提出してもらう必要はありません。来週の授業で答え合わせをします。」と注意を書いていました。

では、答え合わせをしてみましょう。

命題論理言語と言われる記号言語への上の 5 つの日本語の翻訳文を与えるのがこの問いでした。定義 3.2 は論理式の定義と呼んでいますが、前回の解説で注意したように、「論理式」という専門用語は普通の言葉で言えば命題論理言語の「文」ということですので、定義 3.2 は、どのような記号の列が(この命題論理言語における)文であるかを判定する「文法規則」を与えていることになります。

つぎのような解答例を考えてみましょう。

(i)　　太郎が来れば花子は嬉しい。
この翻訳は、
$P \rightarrow S$
(ii)　　太郎も次郎も来れば花子は嬉しくない。
$(P \wedge Q) \rightarrow \neg S$
(iii)太郎か来るかまたは次郎が来るかまたは三郎が来るかのいずれかであれば、花子は嬉しい。
$(P \vee Q \vee R) \rightarrow S$

(iv)太郎と三郎のうちの一人だけが来れば花子は嬉しい、そして、太郎と三郎の両方が来れば花子は嬉しくない
$((P \vee R) \wedge \neg(P \wedge R) \rightarrow S) \wedge ((P \wedge R) \rightarrow \neg S)$

(v) 三郎が来れば花子が嬉しいというようなこは(決して)ない。
$\neg(R \rightarrow S)$

この解答、いかがでしょうか。

さあ、ここで受講者のあたなが先生になったとして、上の解答を採点してください。
。
定義 3.2 に従った文（論理式）の翻訳になっていたらその点を 1 点として、5 点満点でこの解答者に
あなたはうえの解答に何点与えますか？

ここで一息ついて、定義２．１の文法（論理学における文である「論理式」の生成規則）を他の表現の文法に変えてみましょう。
ここでは文（命題論理の論理式）の代わりに、定義 2.1 と同様の形で、算術表現の文法規則を与えてみましょう。
定義 3.1 の単語帳で、
命題変項 P,Q,R,⋯　（真偽の値が変化する文。）
命題定項　T　（いつでも”真“という定まった値を持つ）「真である」を表す文。
命題定項　⊥　（いつでも”偽“という定まった値を持つ）「偽である」を表す文。
２引数の演算子（文の生成という観点では、文と文をつなぐ論理的「接続詞」と呼びます）として、
論理記号∧, ∨,→ などを、また 1 引数の演算子として¬を用いました。

では、この定義 3.2 に対応する子息算術表現のための単語帳はどのようになるでしょうか。

変項、定項は、算術表現では、変数、定数うと呼ばれるのが普通です。
変数記号として $x,y,z,x_1,x_2,...$
定数として 0,1,2,3,4,5,6,7,8,9,10,11,12,⋯..
（論理の時の定項（値が定まった単文）は２種類だけでした。これは論理の値は“真”か“偽”の一方に限るということを前提としているからです。一方、自然数上の算術の値は無限にあります。このコントラストに注意してください。
いま、問題の単純化のために、自然数上の算術表現に限定し、
２引数の演算子として、足し算＋と掛け算 x だけを考えることにします。

この時、定義 3.2 に対応する算術表現の生成規則は、
整数上の足し算掛け算の表現の文法として次のように表せることになります。

（定義 3.2 に対応する、正しい算術表現の生成規則）

1. 変数 x,y, z,…のそれぞれは正しい算術表現である。
2. 定数 0,1,2,3,…,のそれぞれも正しい算術表現である。
（命題論理言語の命題定項はに二種類だけだったのに対して、整数値は無限にあるので、定数記号も無限個あると考えるのが一つの考え方です。）

3　もし t が算術表現であるとわかれば、(-t)も算術表現である。

4　もし s と t とが正しい算術表現と分かれば、(s+t),(sxt)も正しい算術表現である。

5　以上で正しい算術表現と分かったものだけが（現在考えている）算術言語の正しい算術表現である。

引き算と割り算を加えることもできます。
単純な電卓のプログラムでもこのような算術表現についての生成規則（文法）が必要となることは容易に想像できると思います。

さて、先ほどの採点結果を見てみましょう。

あの解答は 5 点満点で何点でしたか。

正解：　0 点でした。
まず、1 問目ですが、

(i)　　太郎が来れば花子は嬉しい。
この翻訳が、
P → S
でしたが、2 引数であっても、1 引数であっても、論理記号（論理的接続子）を伴う複合論理式は引数を入れるたびにカッコが足される生成規則を定義 3.2 では採用しています。定義 3.2 の(３)と(４)を確認してください。
ですので、定義 3.2 の(4)より、正しくは
(P → S)
が命題論理の言語への翻訳文だということになります。
よって問 1 の解答は 0 点です。

同様に翻訳文が単文（命題変更か命題定項）でないかぎり、いちばん外側には同様の理由で

カッコが入っているはずです。

しかし、ほかの問題の解答をみるとどれもいちばん外側にカッコが入っていません。
よって、以下のすべての解答は定義 3.2 に反していることが分かります。
。
P → S
いま、この一番外側のカッコの付け忘れについては大目に見てあげることとして、そのうえで、他に文法違反を犯していないかをみてみることにしましょう。(つまり、いま、いちばん外側のカッコについての使忘れについて以外にも問題点があるかどうかを見てみましょう。)
XXXXXXXXXXXXXXXXXXXX
(i)　　太郎が来れば花子は嬉しい。
(P → S)
このように外側のカッコが入れば正解。

(iii)　　太郎も次郎も来れば花子は嬉しくない。
((P∧Q) →¬ S)
この表現ではまだ定義 3.2 の定義通りでありません。定義 3.2 の 3 で一引数の否定記号導入の時にも
カッコをいれるのが私たちの従うべき文法規則なので、もう一つかっこが入って、
((P∧Q) →(¬ S)) すすれば正解となるわけです。
(iii)太郎か来るかまたは次郎が来るかまたは三郎が来るかのいずれかであれば、花子は嬉しい。
((P ∨Q ∨R) → S)
論理記号∨は 2 引数のカッコつき演算子（論理的接続詞）なので、ここは、
さらにかっこを付けて、
(((P ∨Q) ∨R) → S)と翻訳するか、((P ∨(Q ∨R)) → S)と翻訳する必要があります。
この章のすぐあとで学習する命題言語の意味論の章でわかることですが、どちらの表現も意味論的には同じ意味であることが分かります。

(iv)太郎と三郎のうちの一人だけが来れば花子は嬉しい、そして、太郎と三郎の両方が来れば花子は嬉しくない
(((P∨R) ∧¬(P∧R) →S) ∧((P∧R)→¬ S))
まず、定義 3.2 の(iii)で¬を入れるときにカッコがいりますので、それを入れると、
(((P∨R) ∧(¬(P∧R)) →S) ∧((P∧R)→(¬ S)))
となります。しかしこれだけではまだ命題論理言語の文法通りの文（論理式）ではありませ

ん。
前半に、(A∧B→C)というパターンの部分がありますが、ここだけを見てもまだ文法通りでないことが分かります。
∧と→のどちらが先に定義 3.2 をどう適用した構文構造になっているかが不明です。
この日本語問題文の翻訳の場合は、(A∧(B→C))のほうではなく、((A∧B)→C)の方をとらないと問題文の翻訳になりません。
ですので正解は、
((((P∨R) ∧(¬(P∧R))) →S) ∧((P∧R)→(¬ S)))
ということになります。

(v) 三郎が来れば花子が嬉しいというようなこは(決して)ない。
(¬(R →S))
これは、いちばん外側のカッコが定義 3.2 の（iii）の適用ににり入れば、これで正解です。

いずれにしても、文法規則は厳密にあいまいさがないように規定される必要があることを確認しておきましょう。たとえばプログラミング言語のコンパイラはプログラムの機械語への翻訳時にプログラムに文法違反がないかどうかをチェックするわけですが、この時もプログラミング言語の文法規則が厳密に規定されていることが前提となっています。

——————————————————————————————

３．カッコの省略の約束の導入について

私たちが採用している命題論理言語の文（論理式）の文法（生成規則）では、文の構造上のあいまいさがないようにするためにあらゆる演算子（論理的接続子）の適用ごとにカッコをいれています。

例えば、「計算論理学の単位を取得する　かつ　情報処理入門の単位を取得する　またはプログラミング演習の単位を取得するととる」
ということが 2 年生から 3 年制への進級条件だったとしましょう。
このような進級条件の表示では対象学生は困ってしまいます。どちらにカッコが入っているかによってこの論理的複合文の意味が変わってきます。つまり、ここには意味のあいまいさがあるわけです。
文法構造に依存して意味や真偽値が変わるので、文法規則ははあいまいさの余地のないも

のにしておく必要があります。
他方で、文法規則は文法的チェックの処理にためにできりだけ単純な規則である必要があります。

定義 3.2 の文法も、それに対応する算術表現の文法も「文脈自由文法」と呼ばれる単純な生成規則の形をしています。
この教材では定義 3.2 を命題論理の正式な文法規則としたうえで、かっこを省略してもあいまいさが失われない範囲でかっこを外す略記法の約束事（規約）を指定するという方法をとることにします。

略記法の規約については、基本教材の 8 ページ目後半にあります。これはあくまでも略記法についての約束であり、正式な文法規則（文の生成規則＝定義 3.2）とは区別してください。

略記の約束の 1．いちばん外側のカッコは常に省略できる。

略記の約束の 2．　基本教材 8 ページ目のボックスで示すように、論理的接続詞間の結びつきの強さを指定し、より強い結びつきの接続子とより弱い結びつきの接続子が隣に並んで現れているとき、強い結びつきに接続詞に配所のカッコが入る場合についてのみ、カッコは省力できることと約束する。、
1 引数の“否定”が一番強く、
“かつ”と”または“　が 2 番目に強く、
“ならば”と“同地”は一番結びつき方が低い、
という優位関係を約束事として導入しています。
8 ページの下のボックスを参照ください。
結びつき方関係を導入してカッコを省略するというやり方は他の文法表現でもよく出会います。

例えば、上で導入した算術表現の文法を考えてみましょう。
いま、引き算記号、割り算記号、1 引数のマイナス記号なども入れた四則計算言語を考えてみるとき、
マイナスの結びつき方が一番強く、次がかけ算、割り算、最もくっつき方が低いのが足し算、引き算
とし,、いちばん外側のカッコは省略できて、さらに、
すべて気カッコをいれた文法表現として、例えば、
((3x5)+(-(4+(2x-6))))

にたいして、
3x5+-(4+2x-6)
と省略できることになります。
——————————————————————

４．補足　中置記法と前置記法（評価 S を目標にしている人向きの補足と問い：ただし提出対象ではありません）

上の解説３の答え合わせ部分の(iv)の正解は、
((((P∨R) ∧(¬(P∧R))) →S) ∧((P∧R)→(¬ S)))　　(i
ですた。
私たちが基本教材７ページの文法規則では、２引数の演算子（論理的接続詞）を適用するときには、その接続詞を真ん中において、第１引数を左に、第２引数を右において、たとえば、(A∧B)のようにカッコつきで表示するものでした。そうでないと構文の構造にあいまいさが生じるからでした。
このような演算子と引数の位置関係の表示法は一般に中置記法と呼ばれます。通常に日本語や英語や算術の表現でも中置なので（例えば、Taro comes and Jiro Comes)、中置記法は通常の日常言語や数学の記法とも一致していることから、ユーザーフレンドリーな記法だと考えられています。一方で、演算子を２引数の前におく記法があります。たとえば、(A∧B)の代わりに、
∧AB と表記します。∧だけでなく、他の論理的接続詞（論理記号）についても中置記法を採用することにします。

問Ａ　このような前置記法の表現に中置記法の文（論理式）((((P∨R) ∧(¬(P∧R))) →S) ∧((P∧R)→(¬ S)))　を変換してみてください。
問 B　ここで得られた前置記法の表現はカッコを一つも使わないにもかかわらず、文法的あいまいさが生じていないことを確かめてください。一つの確かめ方と例としては、この前置記法の文を中置記法の文へ変換する手続きが一意的に定まることをみることでもあられます。やってみてください。
問Ｃ　前置記法の論理式（文）を生成する文法規則にするには、定義 3.2 をどのように変更すればよいでしょうか。定義 3.2 の形をなるべく変えないで（最小限の変更で)、前置記法の文法規則の定義を作ってください。
問Ｄ　なぜ前置記法の文法に従うと、カッコをつかわないのにもかかわらず、一意的に（曖昧さなしに）構文構造（文法構造）が確定するのかについて説明できるか考えてみてください。
問Ｅ　中置記法の文（論理式）の入力に対して、前置き法の文（論理式）が出力するような

プログラムの疑似コード（またはなんらかのプログラム言語上でのプログラム）を作成してみてください・

——————————————————————————

５．最上級者向けのクイズ

（ここはご自由にスキップしていただいて、次の５に進んで結構です。 これに答えなくても S 評価を取得るることができます。） 提出の仕方は 10 月 13 日に「お知らせ」に掲示します。

最上級者用のクイズ課題（よくできているとボーナス点を差し上げます）：
ボーナス点の対象問題です。特に興味をもったひとや、プログラム言語理論、特に形式言語理論などに興味を持っている人用の問題です。（S を取得するのにこの解答は必要ではありません。）
「文脈自由文法」という言葉を出しました。基本教材の定義 3.2 も上に挙げた算術表現文法もその典型です。
基本教材の８ページ目と上で、カッコの省略のための規約を、省略法として定義 3.2 を前提として定めました。
実は、ちょうど８ページのようにカッコを省略した形のものが直接論理式として生成できるような文脈自由文法も作れます。
「文脈自由文法」とは何かを調べたうえで、
１．定義 3.2 の文法が確かに文脈自由文法になっていることをせｒス名してください。
２．定義 3.2 プラスカッコ省略規約のかたちではなく、直接このカッコ省略形が生成されるような文脈自由文法を書いてください。
これが、ボーンス点のための課題です。「いちばん外側のカッコは省略できる」という文を文字通りに受け取ると、その表現が一番外側かどうかを確認してから、そうであればカッコを消す、という文脈的な要素が規則適用の条件に入るようにも見えます。ですので、一見すると、文脈自由文法で書けるのかどうかはっきりしませんが、じつはすこし工夫すれば文脈時 tyy 文法で書けます。）

（このようなレベルの高い人のためのクイズ課題を時々出します。

６．基本教材「第４章　命題論理の意味論」についての補足

意味論と言ってもいろいろな種類の意味論があります。この授業ではまず「表示的意味論」(Denotational Semantics)と呼ばれる意味論を「命題論理言語」というとても限られた範囲内で学習します。後ほど、型付きラムダ計算論の意味論についても補足します。(基本教材第６節の補足説明として解説します。) また、順調に進めば学期末になってから、(命題論理言語を拡張してつくられる)「述語論理言語」の意味論についても表示的意味論の立場から紹介します。
プログラムンぐ言語理論においても、プログラムの正しさについて議論するときに取り扱われる意味論の考え方の一つがこの「表示的意味論」です。自然言語意味論でも「形式意味論」という名前で表示的意味論が使われています。

実際にこの第４節で取り扱われるのは、真理表による論理式（命題論理の文）の真偽の計算です。ですので、総合教育科目「論理学」やその他の基礎科目でも真理表ならすでに勉強した、という人もいると思います。
ただし、真理表を「表示的意味論」の一例だと意識しながら学習することはあまりないと思います。第 4 節のテキストの説明のの仕方は表示的意味ということを強調した書き方になっています。

しかし、基本教材のこの第３章の例題と練習問題が解けるのであれば、特に現段階で「表示的意味」について深い理解は必要ありません。第３章を読み進んでいければ、以下の説明はスキップしていただいても結構です。表示的意味についての補足的説明は以下でも言及しているとおり、第５章でも行います。

なお、下の表示的意味論についての補足説明とはべつに、すべての受講者の方たちに向けた「問い」を設定しておきます。
真理表だけをもとにして例題や練習問題を解いていくのはほとんどの受講生にとって簡単な作業だと思いますので、多少刺激になる問いを出しておきたく思います。
この問いも提出の必要はありません。来週の第３回目の授業で答え合わせをしたく思います。

問１.「ならば」(If…then….）の真理表をどう思いますか。
どこか不自然だとかんじるところはありますか。
もしあれば、この真理表に従って真偽値を与えると不自然となる「A ならば B」の形の例文を日本語でみつけて、なぜ不自然であるとおもうかを説明してみてください。ファイルに書き留めておいてください。
さらに、この「ならば」の真理表の不自然さを解消するには、この「ならば」の真理表のどこをどう変えたらよいかを提案してください。

問２　この真理表により論理式の真偽の計算は、計算量的爆発を起こしてしまいます。ですので、このような表のような真偽値の評価法は現実的・実践的評価法とは言えず、計算標的爆発を起こしてしまう応用領域では、実践にはいろいろさらなる工夫が必要となると考えられます。ここで、「この真理表により論理式の真偽の計算は、計算量的爆発を起こす」ということは何を意味しているかを答えてください。

７．上級者向け：真理表を用いる意味論の前提としての「表示的意味論」のコンセプトの入門

命題倫理言語は最も単純な論理的言語であるため、言語が単純すぎることにより「表示的意味」ということがかえってわかりにくいかもしれません。ここではまず述語論理言語と呼ばれる言語で「表示的意味」というコンセプトを紹介し、その特別な場合として命題論理の表示的意味を説明することにします。

「太郎は学生だ」は単文で、真であったり偽であったりする命題（文）であるという意味で、命題論理言語においては、命題変更 P,Q,R などで記号的に翻訳できるものです。一方、述語論理言語という命題論理を拡張した論理言語では、この文の守護部「太郎」と述語部「＊は学生だ」を区別（分節化）して表現します。（第７章参照）
「学生だ（＊）」を１引数述語記号と呼び、＊は名前を入れられる部分とそます。この一引数述語「学生だ(＊)」に「太郎」を代入して「学生だ（太郎)」という文（論理式）がつくられます。日本語に翻訳するときには「太郎は学生だ」と翻訳することにします。
今、表示的意味論の基本的な考え方をこの論理式の意味の与え方（解釈の仕方とも言います）によって示すことにします。

言語表現と対象の世界とを区別して、言語表現が対象世界の何を指しているかについての指し示す関係（言語表現から対象への矢印＝reference）のことをその表現の意味（または意味解釈）とみなします。
「みつ」はわたしの名ですが、この「みつ」自体は無意味な言語表現でしかないという立場をとります」。
いま言語表現「みつ」に対して世界にいる私（この科目の担当教員の私）が意味論的に指し示される関係にあるとすると、すの指し示しによって「みつ」という言語表現が世界の側の私を表示しているとされ、この指し示しが一つの意味解釈を与えているということになります。

「太郎」だとか「学生だ（＊)」は言語の側の無意味な表現です。言語表現の表示的意味とは言語表現が対象世界の側のどんな対象を指し示すかという指し示し方のこととします。「学生だ（＊)」という言語表現が指し示すのが（言語の外の）世界の側の「学生の集合」だると通常私たちが理解しているとおりの「学生だ」の表示的意味だということになります。一方で、「太郎」という名前（固有名詞）の表示的意味解釈とは、「太郎」という名前が世界の何を指しているかすることにあります。この指示（指し示す）関係を（表示的）意味解釈と考えます。
そのような標準的な「学生だ」の表示的意味解釈のもとでは、「太郎」の指し示している対象（太郎の表示的意味）が実際に「学生である」の指し示す表示的意味である集合（標準解釈では、世界の側の学生の集合）のメンバーであれば、「学生（太郎)」という文の指し示している（表示的）意味は「真」(つまりこの文は世界の側の「真」を指し示していることとなります。「太郎」の指し示している世界の対象（「太郎」の表示的意味）が「学生だ」の指し示している表示的意味（ここでは、世界の側の“学生の集合”）のメンバーでないならば、「学生だ（太郎)」という文は世界の側の「偽」という対象を指し示している、と考えます。

ここで、世界に「真」そのものとか「偽」そのものといった抽象的（または、概念的）対象が実在しているのか、と疑問に思う人も入りでしょうが、表示的意味論の考え方では、抽象的対象も世界の中の対象と考えます。このことによって、ある言語表現に対する対象の指し示し方だという考え方に一貫性を与えることになります。上で導入した算術表現に対しても世界の側に抽象的数が対象としてあると考えて表示的意味論を一貫させます。

ところで「学生である（＊)」の表示的意味が世界の“先生の集合”を指しているという意味解釈の下では、「みつ」の指し示している意味解釈が担当教員わたしであるとき、「学生である（みつ)」(つまり、日本語に翻訳すると、“みつは学生である”）は「真」を指し示していることになります。このように、「学生である」や「みつ」の指し示し方、表示的意味解釈の仕方により文の主語部や述語部の意味解釈が変わり、言語表現としては同じ文であってもその文が指し示す真偽値は変わります。文の各部分のどのような表示的意味解釈の違いによっても文全体は真を指し示す、というような文のことをトートロジー（恒真）な文と呼びます。第 4 節の命題論理の意味論では、基本的にはこのような意味でのトートロジーについて学習します。
命題論理言語では主語部と述語部を分けて（分節化して）それぞれ独立に表示的意味解釈を与える、ということはありません。言語表現の最小単位は単文（命題変更または命題抵抗）です。命題抵抗 T は常に真を指し示すので抵抗と言われます。一方、命題変更という単文、（記号で P,Q,R などで表すもの）は真を指し示したり、偽を指し示したりする表示の指し示し方の自由度があるわけです。命題論理言語では表示的意味解釈の自由度があるのは命題変更の表示的解釈のみであり、しかも指し示せる世界の側の対象は「真」そのものと。「偽」

そのもの、の 2 つしかありません。現れる複数の命題変更の表示的意味解釈が与えられたら、あとは文（論理式）の文法構造だけによって、複合文全体が「真」のほうをさししめすのか、「偽」であるかが決定します。これは論理的接続詞（論理記号）に表示的意味解釈が一定にきまっているからです。現れる単文ごとのの意味解釈（真を指し示す場合と偽を指し示す場合の組み合わせ）がすべて決まれば、論理的結合子の解釈に従って表を作って計算していけば、現れる単文の真偽の取り方によって（解釈の仕方によって）複合文全体がどのような審議の指し示し方をするかが決定できるわけです。部分部分の真偽値が決まると、文の文法的構造に従って、より複雑な複文の真偽の指し示し方も決定できるということを「意味の合成原理」と言います。表示的意味論における基本的な考え方です。

———