

# 言語処理 (Formal Languages Processing)

本科	選択・必修	開設時期	単位数	授業形態	担当
情報電子	選択	5年	2	講義	江口 賢和

## 【授業の概要】

プログラミング言語処理系の論理的構造や各種アルゴリズム、及びその作成のための基本的手法を理解させる。さらに、言語処理の基礎としての言語の構文解析理論やその適用法について理解させる。

## 【授業の進め方】

座学の講義を基本に、演習、実習を交えながら進める。また、適宜レポートを課す。最後に、lex/yacc を用いて簡単なコンパイラを作成する実習を行う。

【授業の概要】	【授業項目】	【内 容】
1回	言語処理の概要、処理系、表記法	言語処理の概要、その処理方式などについて学ぶ。また、プログラムや処理系など表記法を導入し、処理系の構成法などを学ぶ。
2回	コンパイラの構造	コンパイラの論理的な構成、物理的な構成について学ぶ。
3回	文法と言語	形式言語理論（チョムスキー理論）およびオートマトン理論の概要を学び、プログラミング言語の意味を理解する。
4回	コンパイラ言語の構文	B N F による定義が形式言語のどの範疇に入るのかを示し、これを使ってコンパイラ言語の構文が定義されることを学ぶ。
5回	字句解析（正規表現と有限オートマトン）	字句の構文が正規表現に当たり、正規表現の処理は有限状態オートマトンになること学び、字句解析が有限状態オートマトンの構成に対応することを理解する。
6回	字句解析（有限オートマトンの構成、字句解析器生成系 lex）	UNIX ツールである字句解析器生成系 lex を取り上げ、字句解析ルーチンが自動的に生成可能であることを学ぶ。
7回	lex による字句解析ルーチンの作成 I	簡単な字句を B N F で定義し、その字句を解析するルーチンを lex を用いて作成する実習を行う。
8回	中間試験	今までに学んだ内容から問題を出題し、理解度をみる。
9回	中間試験の解説および lex による字句解析ルーチンの作成 II	中間試験の解説および lex 実習の続きを行う。
10回	構文解析（上向き構文解析、下向き構文解）	構文解析の2つの範疇である上向き構文解析（bottom-up method）と下向き構文解析（to-down method）について学ぶ。
11回	構文解析（演算子順位解析）	上向き構文解析の代表例である演算子順位解析について学ぶ。
12回	演算子間の優先順位を求める実習 I	演算子間の優先順位を求める方法を述べ、与えられた生成規則から優先順位を求めるプログラムの作成実習を行う。
13回	演算子間の優先順位を求める実習 II	引き続き与えられた生成規則から優先順位を求めるプログラムの作成実習を行う。
14回	構文解析（LL(1) 構文解析）	下向き構文解析の代表的な文法である L L ( 1 ) 文法を取り上げ、下向き構文解析の特徴及び問題点について学ぶ。
	期末試験	中間以降、今までに学んだ内容から問題を出題し、理解度をみる。
15回	解答返却など	定期試験の解答を行う。
16回	構文解析（LR 構文解析の概説）	上向き構文解析のもう一つの代表例で、かなり強力な構文解析である L R 構文解析について学ぶ。
17回	構文解析（SLR(1) 構文解析）	L R 構文解析の中で最も単純な解析法である S L R ( 1 ) 構文解析について学ぶ。
18回	構文解析（LR(1)、LALR(1) 構文解析）	L R ( 1 ) 構文解析及び解析表を小さくした L A L R ( 1 ) 構文解析について学ぶ。

19回	構文解析器生成系 yacc システム	LALR(1) 構文解析ルーチンを自動生成するUNIXツール yacc について学ぶ。				
20回	yacc によるインタプリタの作成 I	yacc を使って簡単な言語のインタプリタを作成する実習を行う。				
21回	yacc によるインタプリタの作成 II	引き続きインタプリタの実習を行う。				
22回	意味解析 (文法の構造)	ブロック構造等に関連して意味解析について学ぶ。				
23回	中間試験	後期で、今までに学んだ内容から問題を出題し、理解度をみる。				
24回	中間試験の解説及び意味解析 (記号表の管理 1)	中間試験の解答を行う。記号表の構造や管理法について学ぶ。				
25回	意味解析 (記号表の管理 2) エラー処理	記号表の構造や管理法について学ぶ。 エラー処理 (エラーの発見、エラーからの復帰等) について学ぶ。				
26回	意味解析 (実行時の記憶域の管理)	各種データ構造に対する領域の割り当て法や実行時の領域の管理法について学ぶ。				
27回	意味解析 (コード生成)	対象マシンの特徴、及び各制御構造に対するコード生成について学ぶ。				
28回	yacc による簡単なコンパイラの作成 I	yacc を使って簡単なプログラミング言語のコンパイラを作成する実習を行う。				
29回	yacc による簡単なコンパイラの作成 II	引き続きコンパイラ作成の実習を行う。				
	期末試験	実施せず。				
30回	解答返却など	コンパイラ作成の実習の解説を行う。				
【到達目標】	形式言語の定義、翻訳系の構造 (字句解析、構文解析、意味解析を理解) を説明でき、具体的な例としての簡単なプログラミング言語の処理系 (インタプリタ、コンパイラ) を作成することができる。					
【徳山高专学習・教育目標】	A 1	【JABEE 基準 1(I)】 d-1				
【評価法】	定期試験 (60%)、実習レポート (30%)、学習状況 (レポート、出席率、受講態度等) (10%) によって総合評価をする。					
【テキスト】	参考図書: 中田育男「コンパイラ」(産業図書) 中田育男「コンパイラ」(オーム社) 五月女健治「yacc/lex」(啓学出版)					
【関連科目】	プログラミング言語 (2年)、情報数学 (3年)、アルゴリズムとデータ構造 (3年)					
【成績欄】	前期中間試験	前期末試験	前期成績	後期中間試験	後期末試験	学年末成績
	【     】	【     】	【     】	【     】	【     】	【     】