

## 2026 年度シラバス

科目分類/Subject Categories			
学部等/Faculty	/工芸科学部 : /School of Science and Technology	今年度開講/Availability	/有 : /Available
学域等/Field	/設計工学域 : /Academic Field of Engineering Design	年次/Year	/3年次 : /3rd Year
課程等/Program	/情報工学課程・課程専門科目 : /Specialized Subjects for Undergraduate Program of Information Science	学期/Semester	/前学期 : /First term
分類/Category	/:/	曜日時限/Day & Period	/水 4 : /Wed.4

科目情報/Course Information				
時間割番号 /Timetable Number	12212302			
科目番号 /Course Number	12260019			
単位数/Credits	2			
授業形態 /Course Type	講義 : Lecture			
クラス/Class				
授業科目名 /Course Title	コンパイラ : Compiler			
担当教員名 / Instructor(s)	/水野 修 : MIZUNO Osamu			
その他/Other	インターンシップ実施科目 Internship	国際科学技術コース提供科目 IGP	PBL 実施科目 Project Based Learning	DX 活用科目 ICT Usage in Learning
				○
	実務経験のある教員による科目 Practical Teacher			
科目ナンバリング /Numbering Code				

授業の目的・概要 Objectives and Outline of the Course	
日	プログラムを作成しようとするときに必要なコンパイラを対象とする。本講義の目的は、形式言語理論の基礎、及び、コンパイラの基本構造とそこで用いられているアルゴリズムを理解し、その結果として、コンパイラを自分で作成できるようになることである。コンパイラは代表的な言語処理プログラムであるので、本講義を通じて一般的な言語処理手法を学ぶことができる。
英	Compilers are the programs which transform source programs in high-level languages to target programs in machine languages. This lecture presents the introduction to formal language theory, basic structure of compilers, and some algorithms which are used in compilers.

学習の到達目標 Learning Objectives	
日	<p>コンパイラの基本的構造を理解し、説明できる。</p> <p>正規表現と有限オートマトンを理解し、正規表現から同じ言語を受理する有限オートマトンを構成できる。</p> <p>文脈自由文法や EBNF 記法などを理解し、それらで書き表された構文を理解できる。</p> <p>LL 構文解析法を理解し、LL 構文解析系を作成できる。</p> <p>LR 構文解析法を理解し、SLR 構文解析表を作成できる。</p> <p>型の等価性、スコープルールや引数の渡し方などの概念を理解し、その実現方法を説明できる。</p> <p>実行時環境を理解し、説明できる。</p> <p>プログラミング言語の文や式の意味を理解し、コード生成法を説明できる。</p>
英	<p>Able to explain fundamental structure of compilers.</p> <p>Able to translate from a given regular expression to the finite state automaton which accepts the language defined by the</p>

expression. Able to explain the syntax in the context-free grammar and EBNF notation. Able to write LL (1) parser in some programming language. Able to explain the LR syntax analysis methods and make SLR (1) syntax analysis table. Able to explain the concepts of type equivalence, scope rule, and parameter passing methods. Able to explain the run-time environment. Able to explain the meaning of statements and expressions in the programming languages and the code generation method.
--

学習目標の達成度の評価基準 / Fulfillment of Course Goals (JABEE 関連科目のみ)	
日	
英	

授業計画項目 Course Plan			
No.		項目 Topics	内容 Content
1	日	コンパイラとは	コンパイラの役割とコンパイラの基本的構造。
	英	What is the compiler?	the needs of compilers and the fundamental structure of the compilers
2	日	形式言語(1)	形式言語、正規表現と正規言語。
	英	Formal languages (1)	formal languages, regular expressions, and regular languages
3	日	形式言語(2)	有限オートマトン。
	英	Formal languages (2)	finite state automata
4	日	形式言語(3)	文脈自由文法と文脈自由言語。BNF 記法と構文図。
	英	Formal languages (3)	context-free grammars, context-free languages, BNF notations, and the syntax chart
5	日	字句解析	字句解析系。正規表現から決定性有限オートマトンへの変換。
	英	Lexical analysis	Lexical analysis
6	日	構文解析(1)	下向き構文解析。左括り出しと左再帰性の除去。LL 構文解析。
	英	Syntax analysis (1)	LL (1) syntax analysis method
7	日	構文解析(2)	上向き構文解析。SLR 構文解析。
	英	Syntax analysis (2)	SLR (1) syntax analysis method
8	日	構文解析(3)	正準 LR 構文解析。LALR 構文解析。
	英	Syntax analysis (3)	LR (1) and LALR (1) analysis methods
9	日	型検査	型の等価性。型変換。多重定義。
	英	Type inspection	type equivalence, type translation, and overloading
10	日	記号表	記号表の構成法。スコープルールとその管理。
	英	Symbol table	the structure of the symbol table and scope rule
11	日	実行時環境(1)	実行時スタックと駆動レコード。変数の参照法。
	英	Run-time environment (1)	run-time stack, activation record, access method to variables
12	日	実行時環境(2)	手続き呼び出しと引数の引き渡し。
	英	Run-time environment (2)	calling method to procedure/function and parameter passing methods
13	日	コード生成(1)	制御文のためのコード生成。
	英	Semantic analysis (1)	intermediate code generation for control statements
14	日	コード生成(2)	算術式及び論理式のためのコード生成。
	英	Semantic analysis (2)	intermediate code generation for assignment statement and expression
15	日	目的コード生成	実際のコンピュータのためのコード生成法。
	英	Object code generation	code generation method for machine languages

履修条件 Prerequisite(s)	
日	「プログラミング I」及び「プログラミング II」を履修し、C 言語の機能を深く理解していること。「データ構造とアルゴリズム」を履修していること。並行履修でもよい。本科目の履修は「言語処理プログラミング」の履修のために有用である。
英	Courses students need to have taken courses "Programming I," "Programming II," and "Data structures and algorithms" before.

授業時間外学習（予習・復習等） Required study time, Preparation and review	
日	各回に宿題として課題レポートを課す。この課題レポートを提出するために各レポートについて 30 分から 1 時間必要である。また、毎回復習のために 1 時間程度の学習時間が必要である。その他、期末試験のための学習時間も必要である。
英	About 1 hour is needed for each report required weekly. About 1 hour is needed for reviewing of each week. Some hours for learning are needed before final exam.
教科書／参考書 Textbooks/Reference Books	
日	教科書「コンパイラ, 第 2 版」(辻野嘉宏著, オーム社)。
英	Text book : Y.Tsujino: "Compiler, Second Edition" Ohmsha (2019).
成績評価の方法及び基準 Grading Policy	
日	学期末試験 70%、数回の課題レポート 30%により評価する。合計点が 60 点以上を合格とする。
英	The evaluation of this course will be conducted based on the evaluation of final exam.(70%) and the evaluation of the required reports (30%). The students whose result of the evaluation is 60 points or higher will pass.
留意事項等 Point to consider	
日	3 回生後期演習科目「言語処理プログラミング」履修時に便利な知識となるため、本科目と共に受講することが望ましい。(ただし、履修条件とはしていない)
英	The contents of this lecture is useful in the exercise "Language Processing Programming" in fall semester in 3rd grade. It is better to take this lecture to understand "Language Processing Programming" well. (However this lecture is not mandatory condition of the "Language Processing Programming".