

コンパイラ(Compiler)

担当教員名	三浦 欽也		
学科・専攻、科目詳細	電気情報工学科 情報工学コース 5年 前期 1単位 講義		
学科のカリキュラム表	専門科目 必修科目		
共生システム工学の科目構成表	専門工学科目 専門応用系		
学習・教育目標	共生システム工学	F-1(55%)	G-2(25%) H-1(20%)
	JABEE基準1(1)	(d)(e)(h)	
科目的概要	我々がプログラムを記述する際、人間にとて可読性の高いプログラミング言語(高級言語)を用いるのが普通である。コンパイラは、そのような高級言語で記述されたプログラムを、CPUが解釈実行できる言語(機械語)に変換するためのプログラムである。この科目では、これまでに発達してきたプログラミング言語の構文と意味に関するさまざまな理論と、それらに基づいて開発されてきた、プログラミング言語を機械語に変換するための種々の手法について講義を行う。		
テキスト(参考文献)	(教科書) 湯浅太一:「コンパイラ」、昭晃堂 (参考文献) B. W. Kernighan, R. Pike 著、石田晴久 監訳:「UNIXプログラミング環境(第8章)」、アスキー		
履修上の注意	受講に当たっては、マイクロコンピュータ(アセンブリ言語)、プログラミングII、データ構造とアルゴリズム、離散数学(有限オートマトン・形式言語理論)、またはそれらに相当する科目を修得しておくことが望ましい。		
科目的達成目標	この科目ではコンパイラの理論と実際について解説し、コンパイラの開発に必要な知識を習得するとともに、プログラミング言語とプログラムの実行に関するより深い理解を得ることを目的とする(H-1)。具体的な達成目標は、 [1] 字句解析の理論的基礎と手法を理解すること [2] 構文解析の理論的基礎と手法を理解すること [3] 意味解析とコード生成の手法を理解すること である。これらの手法を学習することにより、実践的なコンパイラ作成の能力を培い、実践的な問題解決能力を醸成する(G-2)。また、それらの手法の理論的基礎を学習することにより、プログラミング言語、及び、そのコンパイラを柔軟に設計する基本的な能力を身につけ、プログラムの実行に関するより深い理解を通して、実践的なプログラミングの能力を高める(F-1)。		
自己学習	目標を達成するためには、授業以外に次の自己学習が必要である。 (1) 授業内容を復習する。 (2) 下の欄に示す課題に取り組む。		
目標達成度(成績)の評価方法と基準	合格の対象としない欠席条件(割合)		1/3以上の欠課
	評価方法: 前期中間試験(40%)、前期期末試験(40%)、課題(20%) 評価基準: 達成目標の各々で修得すべき内容を以下に示す。 [1] 正規表現、有限オートマトン、lexの利用 [2] 文脈自由文法、BNF・拡張BNF、LL(1)構文解析、yaccの利用 [3] 名前の解決、スコープと名前空間、関数呼出し・文・式のコード生成 以上の内容に関して、複数の課題(下記)を課し2回の定期試験で出題する。 上記比率で合算して評価し、100点満点中60点以上の者を合格とする。 1) ソースプログラムと一時ファイルからコンパイラの動作を理解する。 2) 正規表現が表す文字列集合のうち一定の長さ以下のものを列挙する。 3) 正規表現と等価な非決定性有限オートマトン・決定性有限オートマトンを構成し、状態数を最小化する。 4) lexで記述されたプログラムを読解し、指示通り修正・改変する。 5) yaccで記述されたプログラムを読解し、指示通り修正・改変する。 6) サンプルコンパイラのソースコードを読解し、動作を理解する。		
連絡先	miura@mail.kobe-c.ac.jp		

授業の計画・内容	
第1週	コンパイラの概要 コンパイラ(compiler)の理論的なモデルと一般的なコンパイル(compilation)の過程、また、コンパイラの構造について概説する。
第2週	字句解析 1/3 コンパイラの字句解析の理論的基礎として、正規表現(Regular Expression; RE)と有限オートマトン(Finite Automaton; FA)について解説する。
第3週	字句解析 2/3 正規表現で表わされた字句構造を受理する有限オートマトンを構成する方法について解説する。
第4週	字句解析 3/3 状態遷移表を用いた字句解析プログラムの実際、また、字句解析プログラムのlexによる自動生成について解説する。エラー処理についてもふれる。
第5週	文法(形式言語理論) 1/2 形式言語理論、特にプログラミング言語の構文定義で一般的に用いられる文脈自由文法(Context Free Grammar; CFG)について解説する。また、BNF・拡張BNF・構文図式についてもふれる。
第6週	文法(形式言語理論) 2/2 形式言語理論における諸概念：記号列の導出・最左/最右導出・解析木・文法のあいまい性等について解説する。
第7週	構文解析 1/3 再帰的下向き構文解析法、特にLL(1)構文解析について解説する。また、左再帰の問題と、その解決法について扱う。
第8週	中間試験
第9週	構文解析 2/3 LR構文解析について概説し、構文解析プログラムのyaccによる自動生成について解説する。
第10週	構文解析 3/3 あいまいな文法への対処法について解説する。また、エラー処理についてもふれる。
第11週	意味解析 1/2 意味解析の概要と言語中の名前とそれが表わすオブジェクトの対応付け(名前の解決)の方法について解説する。また、スコープと名前空間についても解説する。
第12週	意味解析 2/2 前方参照の扱い、型チェックと型変換について解説する。また、エラー処理についてもふれる。
第13週	コード生成 1/2 具体的な実行環境のモデルを定義し、関数呼び出しに対応するコードの生成について解説する。局所変数等のために記憶領域を割当てる方法についても解説する。
第14週	コード生成 2/2 各種の文や式のコードを生成する方法について解説する。
第15週	実習 コンピュータとyacc・lexを用いて簡単な言語処理系を作成し、ツールを用いたコンパイラ開発について理解を深める。
期末試験	