

# プログラミング言語論

科目ナンバー 3B221  
専門 選択 2単位

盛 拓生

## 1. 授業の概要(ねらい)

本講義ではプログラミング言語に関して、その基本事項である各種プログラミングパラダイムとその計算モデル、プログラミング言語の構文、プログラミング言語の意味論について学びます。

特に、現在最も普及している実用的な手続き型言語であるC言語及びオブジェクト指向言語のJavaについて、計算モデルとしての仮想マシン及び仮想マシン上で実行される中間コードを通してそれらの操作的意味論を理解します。

また、現在注目されている、ラムダ算術を計算モデルとする関数型プログラミング言語、人工知能の分野で利用される論理型プログラミング言語、文字列処理などで利用されるスクリプト言語についても概要を示します。

このように多種多様なプログラミング言語、プログラミングパラダイム、計算モデルを知ることにより取り扱う問題に応じて適切にプログラミング言語を選択できるようになることを目指します。

この科目では、学位授与の方針(ディプロマポリシー)DP4C及びDP4Mに関連する能力を修得します。

## 2. 授業の到達目標

この科目では次のような能力を修得することを目標とします。

学生は、主な計算モデル、プログラミングパラダイム、プログラミング言語の性質を理解し、説明できる。

学生は、ソースコードからマシン語への変換という観点から、プログラミング言語を分類でき、使用目的に合わせてプログラミング言語を選択できる。

学生は、プログラミング言語の定義で用いられる記法と文脈自由文法の関係を説明でき、文脈自由文法による導出過程を解析木で示すことができ、曖昧な文脈自由文法の曖昧性除去の方針を説明できる。

学生は、ノイマン型コンピュータを計算モデルとするRandom Access Machineについて、その動作を説明することができる。

学生は、スタックマシン上における関数/メソッドの再帰呼び出しについて説明できる。

学生は、C言語に代表される手続き型言語の特徴を説明でき、その計算機上での動作、操作的意味論を説明できる。

学生は、Java言語に代表されるオブジェクト指向プログラミング言語の特徴を説明でき、その計算機上での動作、操作的意味論を説明できる。

## 3. 成績評価の方法および基準

中間レポート(50%)、期末試験の成績(50%)。

講義内容、レポート、試験への質問、フィードバック等は原則的にLMSまたはオフィスアワーで対応します。

## 4. 教科書・参考文献

教科書

大山口通夫、五味 弘共著 プログラミング言語論 コロナ社、ISBN-13: 978-4339027044

参考文献

ラビ・セシ著、神林 靖訳 プログラミング言語の概念と構造 ビアソンエデュケーション、ISBN-13: 978-4894717701

## 5. 準備学修の内容

予め、各回の講義資料をLMS上で公開します。

講義資料はPC、タブレット、スマートフォン等にダウンロードするか紙に印刷するなどして、講義中いつでも参照、書き込みできるようにしてください。

授業前にこれらの資料に1時間程度目を通した上で、理解できた点、理解できていない点をそれぞれ把握して授業に臨んでください。

授業後には、30分程度かけてLMS上のテストで理解度を確認し、復習してください。

## 6. その他履修上の注意事項

中間レポートが提出されない場合単位の取得が著しく困難になります。中間レポートは締め切りを守って必ず提出してください。

自主学習支援のためにLMSを利用します。講義資料等はLMS上で公開されているので事前に入手し、講義中はノートをとることよりも、

講義を聞いて理解すること、演習問題を解くことに集中してください。

事前に履修すべき科目は、プログラミング1、プログラミング2です。

同時に履修すべき科目は、データ構造とアルゴリズム、情報科学プログラミング1です。

事後に履修すべき科目は、情報科学プログラミング2、オートマトンと計算理論、計算機アーキテクチャです。

この科目はJABEEプログラムの必修科目で、学習・教育到達目標中項目4-4に対応しています。

## 7. 授業内容

- 【第1回】 プログラミング言語の特徴と分類1  
低水準言語と高水準言語、コンパイラ型言語、インタプリタ型言語、中間言語
- 【第2回】 プログラミング言語の特徴と分類2  
計算モデルとプログラミングパラダイム、チューリングマシン、Random Access Machine(RAM)と命令型言語
- 【第3回】 プログラミング言語の特徴と分類3  
ラムダ算術と関数型プログラミング言語
- 【第4回】 プログラミング言語の特徴4  
ラムダ算術とLISP(概要)、論理型言語としてのProlog(概要)、オブジェクト指向言語とその計算モデル(概要)
- 【第5回】 プログラミング言語の構文  
文脈自由文法とBackus Naur Form(BNF)、コンパイラのフロントエンドとバックエンド
- 【第6回】 命令型プログラミング言語1  
構造化プログラミング、データ型
- 【第7回】 命令型プログラミング言語2  
ポインタ変数、ポインタ変数と線形リスト、手続き、引数結合方法、変数のスコープ

- 【第8回】 プログラミング言語の意味論  
操作的意味論、RAMの操作的意味論
- 【第9回】 Tiny C/Small Cの操作的意味論  
仮想マシン
- 【第10回】 Small Cの操作的意味論  
関数定義、関数呼び出し、関数から値を返す、ポインタ、配列
- 【第11回】 オブジェクト指向プログラミング1  
大規模プログラム開発とオブジェクト指向、オブジェクト/クラス/インスタンス、カプセル化と情報隠蔽、継承、多態性
- 【第12回】 オブジェクト指向プログラミング2  
インタフェース、継承/合成/委譲、コレクションとジェネリクス
- 【第13回】 Javaサブセットの操作的意味論1  
非オブジェクト指向機能
- 【第14回】 Javaサブセットの操作的意味論2  
オブジェクト指向機能
- 【第15回】 まとめと期末試験