

科 目 名(Subject)	コンピュータサイエンス I (Computer Science I)		
単 位 数(Credits)	2 単位	開講時期	前期
担当教員名 (Name)	三谷 和史 (Kazufumi Mitani)	研究室番号 (Office)	455
Office Hours	mit@res.otaru-uc.ac.jp まで連絡のこと		
1. 授業目的・方法(Course objective and method)			
目的: 本学の学部教育で使用するプログラミング言語は, それを実行するコンピュータの動作原理であるフォンノイマン型に基づいた手続き型の言語である. しかし, それ以外のパラダイムに基づく言語も存在する. その中でも特に関数型のパラダイムに基づく言語を概観し, 実際に関数型言語を使つてのプログラムの作成を通じて, プログラミングへの理解を深める. 言語としては, SML, OCaml, Haskell, Scheme 等の何れかを考えている. 最終的には, 単に問題を関数的に分析できるだけでなく, 効率が良く関数型言語らしい形で分析できる能力を身につける. 更には, 関数型言語で最適化された関数を手続き型言語に戻すことで, 手続き型言語での正しく最適なプログラム作成に資することが出来ることを学ぶ.			
方法: 輪講と演習			
2. 達成目標(Course Goals)			
・問題の構造を再帰的・関数的に理解できること.			
・高階関数を使ったプログラムの作成ができること.			
・関数の最適化を理解できること.			
・遅延評価を使ったストリームの概念を理解し作成ができること.			
3. 授業内容(Course contents)			
学生の興味や進具合によって多少の変更はあるが, おおよそ以下の通り.			
初回: イントロダクション			
2-4 回目: 正格言語 SML と非正格言語 Haskell の概観と基本的プログラミング			
5-6 回目: 高階関数を使ったプログラミング			
7-8 回目: 末尾再帰と CPS			
9-10 回目: SML での無限リストの実現とモジュールシステム			
11-12 回目: Scheme の Call/CC と SECD マシン			
13-14 回目: 関数型言語による言語処理系の作成例について			
15 回目: まとめ			
4. 事前学修・事後学修(Preparation and review)			
課題のプログラム作成, デバッグ及び文献等の精読			
5. 使用教材(Teaching materials)			
適宜指示する. また, プログラミング作成のための問題プリントを配布する.			
6. 成績評価の方法(Grading)			
定期試験は行なわない. 講義への参加と課題への取組状況で総合的に評価する.			
具体には出席率3割, 講義への参加度3割, プログラミング4割程度で評価する.			
7. 成績評価の基準(Grading Criteria)			
社会情報コース標準成績評価基準に従う.			

8. 履修上の注意事項(Remarks)

ある程度のプログラミングを行なえ，プログラミング言語への興味があることが望ましい．英語による文献，本を読む能力があること．