

科目名	オブジェクト指向プログラミング Object Oriented Programming			担当教員	河田 進			
学 年	1 年	学 期	後期	履修条件	選択	単位数	2	
分 野	専門	授業形式	講義	科目番号	14273009	単位区別	学習	
学習目標	オブジェクト指向の各種原理を理解することで、オブジェクト指向の導入がソフトウェア設計の自然なモデル化に有用であることを認識できる。さらに、オブジェクトのデータ構造や振る舞いを設計できるようになり、オブジェクトを使ったソフトウェアを構築できるようになること。							
進め方	教科書を基に、オブジェクト指向の考え方やオブジェクトの設計法を講義し、学習項目毎にC言語やC++言語による課題プログラムをレポートとして作成する。また、実世界に存在する“物”をテーマにオブジェクトを設計する演習を班別に行うが、班全体や分担別の開発状況発表会を適宜行う。							
学習内容	学習項目 (時間数)			学習到達目標				
	1.学習の進め方の説明(1) 2.オブジェクト指向の歴史と必要性(2) 3.構造体と抽象データ型(3) (1)構造体と複素数 (2)抽象データ型とスタック 4.クラスの記述法とカプセル化(6) (1)カプセル化と情報隠蔽 (2)クラス化とインスタンス (3)メソッドとメッセージ (4)コンストラクタとデストラクタ (5)Stackクラスについて 5.汎化と継承(4) (1)汎化と継承について (2)Stackクラスを継承するクラスについて 6.メンバ関数のオーバーライドと多態性 (ポリモアフィズム) (2) (1)複素数クラスの実現と演算子のオーバーライド 7.システムと物の分析演習(4) 8.オブジェクト指向に基づくシステム設計(6) 9.試験の解説(2)			ソフトウェア開発にとってオブジェクト指向が自然なアプローチであることを理解する。 構造体について復習し、複素数やスタックなどを例に、抽象データ型をC言語で実現する方法を習得する。 D2:1-2 C++のクラスによってスタックなどを実現し、Cで作成したプログラムと比較し、その違い(カプセル化)を理解する。さらにクラス、インスタンス、メッセージ、メソッドについて理解する。また、初期化プログラムであるコンストラクタと後始末プログラムであるデストラクタの作り方と使用目的を習得する。 D2:1-4 同種のクラスをより一般化したクラスを定義し、元のクラスが新しいクラスを継承することで効率的なプログラミングが可能であることを理解し、クラス階層を実現できる。 D2:1-2 異なるオブジェクトを同じ命令で操作できるポリモアフィズムの概念を理解し、実現できる。 D1:1 D2:1-2 具体的なシステムをコンピュータ上に構築することを目的に、システム内に存在する”物”を分析し、その性質や振る舞いを机上で設計する。 D2:1-4 D3:3 E2:1-3 設計に基づいてシステムを開発する。 E3:3				
	後期末試験							
	10.試験問題の解答(1)							
	評価方法	試験やレポート、授業中の演習で教育目標D,Eの達成度を計り、達成度を点数化して評価する。						
	履修要件	情報処理Ⅱを修得していること。						
	関連科目	情報処理Ⅱ→ソフトウェア設計論→情報構造論→アルゴリズムとデータ構造						
	教 材	C++クラスと継承完全制覇 矢沢久雄著 技術評論社 http://www.di.takuma-ct.ac.jp/~miyatake/software-eng						
	備 考	オフィスアワー：毎水曜日放課後～17:00						