

熊本高等専門学校		開講年度	令和06年度（2024年度）		授業科目	オブジェクト指向プログラミング
科目基礎情報						
科目番号	HI1413		科目区分	専門 / 選択		
授業形態	授業		単位の種別と単位数	学修単位: 2		
開設学科	人間情報システム工学科		対象学年	4		
開設期	通年		週時間数	1		
教科書/教材	結城浩 著「Java言語プログラミングレッスン 第3版 (下)」 ソフトバンククリエイティブ					
担当教員	神崎 雄一郎					
到達目標						
(1) オブジェクト指向プログラミングの特徴を生かしたソフトウェア実装を, Javaを用いて行うことができる。 (2) クラス図など, UMLのダイアグラムによってソフトウェアの構成を記述する方法について理解し, 説明できる。						
ルーブリック						
	理想的な到達レベルの目安		標準的な到達レベルの目安		未到達レベルの目安	
オブジェクト指向プログラミングの基礎	クラス, 継承, カプセル化, 多態性などについて理解し, 複数のオブジェクト指向プログラミング言語で具体例を説明できる。		クラス, 継承, カプセル化, 多態性などについて理解し, Java言語で具体例を説明できる。		クラス, 継承, カプセル化, 多態性などについて説明できない。	
Javaによる実装	「標準的な到達レベルの目安」に加え, 創造力を生かした応用課題を解くことができる。		クラス, インスタンス, メソッド, フィールド, オーバーロード, オーバーライドなどを含めたJavaの基礎について理解し, 基礎課題を解くことができる。		Javaに関する基礎的な演習課題を解くことができない。	
UML(クラス図・オブジェクト図)	クラス図, オブジェクト図, またその他のUMLのダイアグラムについて理解し, それらを用いてソフトウェアを分析・設計できる。		クラス図やオブジェクト図について理解し, 簡単な記述を行うことができる。		クラス図やオブジェクト図を記述できない。	
ソフトウェア開発の実践	UML等を用いてソフトウェアの構造を自ら設計し, その設計にもとづいた実装を行うことができる。		UMLによって与えられたソフトウェアの設計にもとづき, 実装を行うことができる。		UMLによって与えられたソフトウェアの設計にもとづいた実装を行うことができない。	
学科の到達目標項目との関係						
教育方法等						
概要	オブジェクト指向プログラミングの基礎を身に付けることを目的とした講義である。クラス, カプセル化, 継承, 多態性といったオブジェクト指向プログラミングの基本事項について理解し, それらを生かしたソフトウェア実装方法をJavaのプログラミング演習等を通して体得することを目指す。					
授業の進め方・方法	本科目は, プログラミング演習を中心に進める。成績評価は, 演習課題50%, 定期試験・Webテスト50%の割合で行われる。 演習課題の点数は, プログラミング演習において提出されたプログラムや報告書の内容によって決められる。 定期試験は, 実技 (コンピュータを用いた試験) または筆記試験とする。また, Webテストは, 理解度を確認するための, Webのフォームを用いた簡単なテスト (クイズ) である。					
注意点	この科目は学修単位のため, 30時間相当の自学・自習が求められる。 自学自習の時間は, 講義の復習や, 演習課題の実施などに用いられることを想定する。					
授業の属性・履修上の区分						
☐ アクティブラーニング		☒ ICT 利用		☐ 遠隔授業対応		☐ 実務経験のある教員による授業
授業計画						
		週	授業内容		週ごとの到達目標	
前期	1stQ	1週	プログラミング言語		ソフトウェアの実装に用いる種々のプログラミング言語と設計思想について説明できる。	
		2週	オブジェクト指向プログラミング		オブジェクト指向プログラミングの役割や関連技術について説明できる。	
		3週	Javaの基礎		Javaの特徴や応用分野について説明できる。	
		4週	Java基礎演習 (1)		Javaの基礎文法について理解し, 関連する課題を解くことができる。	
		5週	Java基礎演習 (2)		同上	
		6週	Java基礎演習 (3)		同上	
		7週	Java基礎演習 (4)		同上	
		8週	前期中間試験			
	2ndQ	9週	クラスの拡張・カプセル化 (1)		クラスの拡張, 継承, カプセル化について理解し, 関連する課題を解くことができる。	
		10週	クラスの拡張・カプセル化 (2)		同上	
		11週	クラスの拡張・カプセル化 (3)		同上	
		12週	前期総合演習 (1)		これまでの学習内容に応じた総合的なプログラミング課題を解くことができる。	
		13週	前期総合演習 (2)		同上	
		14週	前期総合演習 (3)		同上	
		15週	前期期末試験			
		16週	定期試験答案返却・解説			
後期	3rdQ	1週	例外処理 (1)		例外処理について理解し, 関連する課題を解くことができる。	
		2週	例外処理 (2)		同上	

		3週	クラスの関係・多態性 (1)	クラスのIS-A関係やHAS-A関係, 多態性について理解し, 関連する課題を解くことができる.
		4週	クラスの関係・多態性 (2)	同上
		5週	UMLによるソフトウェアの設計 (1)	UMLのクラス図等を用いて簡単なソフトウェアの設計を行うことができる.
		6週	UMLによるソフトウェアの設計 (2)	同上
		7週	UMLによるソフトウェアの設計 (3)	同上
		8週	後期中間試験	
	4thQ	9週	抽象クラス (1)	抽象クラスについて理解し, 関連する課題を解くことができる.
		10週	抽象クラス (2)	同上
		11週	後期総合演習 (1)	これまでの学習内容に応じた総合的なプログラミング課題を解くことができる.
		12週	後期総合演習 (2)	同上
		13週	後期総合演習 (3)	同上
		14週	後期総合演習 (4)	同上
		15週	後期期末試験	
		16週	定期試験答案返却・解説	

モデルコアカリキュラムの学習内容と到達目標

分類	分野	学習内容	学習内容の到達目標	到達レベル	授業週
評価割合					
		演習課題	定期試験・Webテスト	合計	
総合評価割合		50	50	100	
専門的能力		50	50	100	